

AFIT/DS/ENY/98-03

**Singularity Avoidance Strategies For Satellite
Mounted Manipulators Using Attitude Control**

DISSERTATION

Nathan A. Titus, B.S., M.S.

Major, USAF

AFIT/DS/ENY/98-03

19980629 027

Approved for public release; distribution unlimited

Disclaimer

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the United States Air Force, the Department of Defense, or the United States Government.

AFIT/DS/ENY/98-03

Singularity Avoidance Strategies For Satellite Mounted Manipulators Using Attitude Control

DISSERTATION

Presented to the Faculty of the Graduate School of Engineering of the Air Force Institute of
Technology Air University In Partial Fulfillment for the Degree of
Doctor of Philosophy

Nathan A. Titus, B.S., M.S.
Major, USAF

Air Force Institute of Technology
Wright-Patterson AFB, Ohio

May, 1998

Sponsored in part by AFRL/VSDVD

Approved for public release; distribution unlimited

Singularity Avoidance Strategies For Satellite Mounted Manipulators Using Attitude Control

Nathan A. Titus, B.S., M.S.

Major, USAF

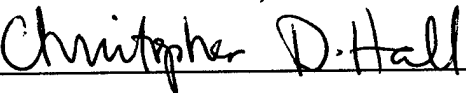
Approved:



Dr. Curtis H. Spenny (Chairman)

4/4/98

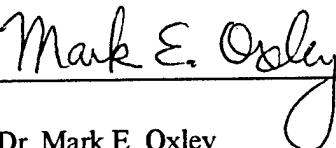
Date



Dr. Christopher D. Hall

5/20/98

Date



Dr. Mark E. Oxley

5/20/98

Date



Dr. Kirk A. Mathews (Dean's Representative)

6/4/98

Date

Accepted:



Dr. Robert A. Calico
Dean, Graduate School of Engineering

Acknowledgments

I would like to thank my research advisor, Dr. Curtis Spenny, for his help and support in this work. His ability to step back and look at the “big picture” proved invaluable in the process of identifying the important areas for study, and his enthusiasm for the subject helped me over the inevitable rough spots in my research. I also appreciate the time and effort given by my other committee members, Dr. Hall, Dr. Oxley, and Dr. Mathews. Dr. Chris Hall provided me with exceptional insights on spacecraft attitude control and was always amazingly responsive to my questions, even after his move to his new position at Virginia Tech. Dr. Mathews must be acknowledged for his extensive (no-holds barred!) review of my work, and his excellent suggestions for future work, including the ideas about the merits a free-floating system with non-zero angular momentum. Dr. Oxley is applauded for his efforts as the committee marathon man, managing to review my work in a timely fashion despite being a committee member for three other doctoral students, all of whom defended within a week of my own defense. Finally, I want to thank my wife, Lori, and children, Emily and Duncan. Lori made many sacrifices for my success, including my absence the day after Duncan was born so that I could take my qualifying examination. Emily and Duncan made the time I could spare from my work enormously satisfying and diverting. Without the love and support of all of them, this accomplishment would not have been possible.

Table Of Contents

Acknowledgments	iii
Table Of Contents	iv
List Of Figures	viii
List Of Tables	xii
List of Symbols	xiii
Abstract	xvii
Chapter 1. INTRODUCTION	1
1.1 Motivation	1
1.2 Research Objectives	6
1.3 Overview	7
Chapter 2. REVIEW OF SMM-RELATED RESEARCH	8
Chapter 3. KINEMATICS & DYNAMICS OF SMMS	13
3.1 Kinematics of Fixed-Base Manipulators	13
3.2 Kinematics of an n -Link SMM	14
3.3 Equations of Motion of an Open Chain Fixed-Base Manipulator	17
3.4 Lagrange's Equations Using Quasi-Coordinates	20
3.5 Equations of Motion of an n -Link SMM	23
3.6 Momenta of an n -Link SMM	32
3.6.1 Linear Momentum	32

3.6.2	Angular Momentum	33
3.7	Summary	36
Chapter 4.	SINGULARITIES OF MOMENTUM-CONSTRAINED JACOBIANS	37
4.1	Free-Floating SMMs: The Generalized Jacobian Matrix and Dynamic Singularities	37
4.2	The Impact Of Redundancy	41
4.3	Prismatic Joints	47
4.4	Joint Limits	52
4.5	A Singularity-Free Design	52
4.6	The Impact of Base Control	55
4.6.1	Full Base Control	55
4.6.2	Base Attitude Control	58
4.7	Summary	65
Chapter 5.	SMM CONTROL	67
5.1	Basic Controller	69
5.1.1	Controller Derivation	69
5.1.2	A Linearization Of The Control Law	73
5.1.3	Stability of the Basic SMM Controller	74
5.1.4	Stability in the Redundant Case	77
5.1.5	A Note on Gain Selection	79
5.2	Free-Floating Case	80

5.3	Base-Attitude Control (BAC) Case	97
5.3.1	Simple BAC Controller	97
5.3.2	Reduced Base Torque Control	100
5.4	Summary	113
Chapter 6.	BASE ATTITUDE CONTROL USING CONTROL MOMENT GYROS	114
6.1	Adding a CMG Cluster to the n -Link SMM Model	115
6.1.1	Dynamic Model	115
6.1.2	SMM Controller Using SGCMG Cluster for Base Actuation	120
6.2	The Effect of CMG Cluster Singularities	122
6.3	Controlling For Singularity Avoidance	128
6.3.1	Avoiding CMG Singularities	128
6.3.2	Avoiding Combined Singularity Conditions (Dynamic and CMG Cluster)	134
6.4	Summary	141
Chapter 7.	CONCLUSION	142
7.1	Conclusions	142
7.2	Recommendations for Future Research	145
Appendix A.	SOME NOTES ON NOTATION	148
Appendix B.	USEFUL IDENTITIES AND PROPERTIES	151
B.1	Differentiating the Base Rotation Matrix	151
B.2	Reducing the Equations of Motion	152

B.3	Passivity property applied to SMMs	153
Appendix C. ELEMENTS OF M AND C MATRICES		155
Appendix D. IMPLEMENTATIONS OF NAKAMURA'S METHODS		158
D.1	Singularity-Robust Inverse (SR Inverse)	158
D.2	Bidirectional Approach to SMM Path Planning	162
Bibliography		166
Vita		169

List Of Figures

Figure 1.	SMM Control Concept Hierarchy	3
Figure 2.	Satellite Mounted Manipulator (SMM)	15
Figure 3.	Defining the position of a link relative to the base center of mass.....	24
Figure 4.	Fixed-Base Planar Arms, Two-Link (a) and Three-Link (b)	41
Figure 5.	Planar SMMs, Two-Link (a) and Three-Link (b)	42
Figure 6.	Inertial Workspace of Two-Link SMM	44
Figure 7.	Inertial Workspace of Three-Link SMM	45
Figure 8.	Planar Two-DOF Arm (RP)	48
Figure 9.	Planar SMM with Two DOF (RP) Arm	50
Figure 10.	Inertial Workspace of Planar SMM with RP Arm	51
Figure 11.	Reducing PDW Using Joint Limits	53
Figure 12.	Planar SMM with RRP Arm	54
Figure 13.	Equivalent Base Motion and End-Effector Motions With Joints Fixed	57
Figure 14.	Two-Link Planar SMM	59
Figure 15.	Two-Link SMM With First-Link Center of Mass Offset	62
Figure 16.	Basic Nonlinear SMM Controller	72
Figure 17.	Basic SMM Controller	74
Figure 18.	SMM Controller with Base-Motion Feedback	82
Figure 19.	A Simple SMM Maneuver (Maneuver One)	83

Figure 20.	Response of Free-Floating SMM to an End-Effector Position Step Input	85
Figure 21.	Torque History for Position Step Input	85
Figure 22.	End-Effector Path for Position Step Input	86
Figure 23.	Circular Path Maneuver (Maneuver Two)	87
Figure 24.	Time History of Distance to Final Position	88
Figure 25.	Torque Requirements for Circular Path (Maneuver Two)	90
Figure 26.	End-Effector Path For Circular Trajectory (Maneuver Two)	90
Figure 27.	Simple Maneuver Prone to Singularity Encounters (Maneuver Three)	92
Figure 28.	End-Effector Paths For Maneuver Three Using Free-Floating Control	93
Figure 29.	Example Path in Joint Space Using Reyhanoglu and McClamroch Control Scheme	96
Figure 30.	End-Effector Paths For Maneuver Three Using Simple Base-Attitude Control And Bidirectional Approach	98
Figure 31.	Total Torque Requirements For Maneuver Three Using Simple Base Attitude Control (Ramp Input) And Bidirectional Approach	99
Figure 32.	End-Effector Path For Position Step (Maneuver One).	105
Figure 33.	Total Torque Requirements For Position Step (Maneuver One)	106
Figure 34.	Base Torque Requirements For Position Step (Maneuver One)	107
Figure 35.	End-Effector Path For Circular Trajectory (Maneuver Two)	109
Figure 36.	Total Torque Requirements For Circular Trajectory (Maneuver Two)	109
Figure 37.	Base Torque Requirements For Circular Trajectory (Maneuver Two)	110
Figure 38.	End-Effector Path For Singularity Prone Position Step (Maneuver Three)	111

Figure 39.	Base Torque Requirements For Maneuver Three	111
Figure 40.	Total Torque Requirements For Maneuver Three	112
Figure 41.	Single Gimbal Control Moment Gyro	116
Figure 42.	Controller for SMM with CMG Cluster	121
Figure 43.	Satellite with Three-Link Elbow Manipulator	123
Figure 44.	Orthogonally Mounted Three-CMG Cluster	123
Figure 45.	CMG Cluster Singularity Measure For Varied Outer-Loop Gain	126
Figure 46.	End-Effector Response For Varied Outer-Loop Gain	126
Figure 47.	CMG Cluster Singularity Measure For Varied Position Step Size	127
Figure 48.	End-Effector Response For Varied Position Step Size	127
Figure 49.	Free-Floating Task Scale Factor vs. CMG Cluster Singularity	131
Figure 50.	End-Effector Response For Controllers With And Without CMG Singularity Avoidance Term	132
Figure 51.	CMG Cluster Singularity Measure For Controllers With And Without CMG Singularity Avoidance Term	133
Figure 52.	Joint Motion Using Controllers With And Without CMG Singularity Avoidance Term	133
Figure 53.	Dynamic Singularity Measure For Varied Outer-Loop Gains Using the CMG Singularity Avoidance Controller	136
Figure 54.	End-Effector Response For Varied Outer-Loop Gains Using the CMG Singularity Avoidance Controller	137
Figure 55.	End-Effector Response Using the SSA (w/ Dynamic Singularity Avoidance) Controller and the CSA (No Dynamic Singularity Avoidance) Controller	140

Figure 56.	Dynamic Singularity Measure Using the SSA (w/ Dynamic Singularity Avoidance) Controller and the CSA (No Dynamic Singularity Avoidance) Controller	140
Figure 57.	Planar Two-Link In Singular Configuration	160

List Of Tables

Table 1.	Two-Link SMM Physical Parameters	43
Table 2.	Three-Link SMM Physical Parameters	44
Table 3.	One-Link (RP) SMM Physical Parameters	50
Table 4.	Controller Gains and Integral Metrics for Linear Trajectory (Maneuver One)	84
Table 5.	Controller Gains and Integral Metrics for Semi-Circular Trajectory (Maneuver Two)	89
Table 6.	Controller Gains For Maneuvers One, Two, and Three	104
Table 7.	Integral Metrics for Maneuver One	108
Table 8.	Integral Metrics for Maneuver Two	110
Table 9.	Integral Metrics for Maneuver Three	110
Table 10.	Physical Parameters for SMM with Three Link Elbow Manipulator	123
Table 11.	Discrete Function For Scale Factors vs.Singularity States	138

List of Symbols

English Symbols

Symbol	Definition
a_i	length of link i
a_i	distance from joint i to body i center of mass
b_i	distance from body i center of mass to joint $i + 1$
c	dynamic singularity measure
d	CMG cluster singularity measure
e, e_1, e_2	error signals in controller
e_{ijk}	alternator function
h	angular momentum
k	scalar gain or scale factor
m_i	mass of body i
p	linear momentum
q	generalized coordinate
q_1, q_2	unactuated (1) and actuated (2) generalized coordinates
r, r_e	end-effector position
v, v_0	spacecraft base translational velocity
C	centrifugal/Coriolis Terms Matrix
D	CMG Cluster Momentum Jacobian
F	potential function
\mathcal{F}_I	inertial reference frame
\mathcal{F}_i	reference frame fixed in body i

H_v, H_ω, H_θ	angular momentum matrices associated with v , ω , and $\dot{\theta}$
I	body Inertia Matrix in principal body frame
J	manipulator Jacobian
J_{fb}	fixed base manipulator Jacobian
J_p	position (translation) portion of manipulator Jacobian
J_a	angular portion of manipulator Jacobian
J_v, J_ω, J_θ	portions of end-effector Jacobian associated with v , ω , and $\dot{\theta}$
K	controller gain matrix
L	Lagrangian
M	manipulator or SMM Generalized Inertia Matrix
P_v, P_ω, P_θ	linear momentum matrices associated with v , ω , and $\dot{\theta}$
Q	generalized force/torque vector
R	rotation matrix
T	kinetic energy
V	Lyapunov potential function
$U, U_{n \times n}$	identity matrix ($n \times n$)

Greek Symbols

Symbol	Definition
ϕ	CMG gimbal angles
λ	eigenvalues
τ	generalized forces/torques
θ	manipulator joint angles
ω	angular velocity

Ω	angular orientation coordinates (quaternions when specifically required)
----------	--

Subscript Symbols

Symbol	Definition
0	pertaining to the satellite base
i	pertaining to body i
v	pertaining to base translational velocity
θ	pertaining to arm joint velocity
ω	pertaining to base angular velocity

Superscript Symbols

Symbol	Definition
T	transpose
#	pseudoinverse
*	Singularity-Robust inverse
*	discrete-like shaped metric
~	label indicating quantity reduced or constrained by momentum conservation

Acronym List

Acronym	Definition
BAC	base attitude control
CMG	Control Moment Gyroscope
CSA	CMG Singularity Avoidance
EOM	equations of motion
GJM	Generalized Jacobian Matrix
NASA	National Aeronautics and Space Administration
PDW	Path Dependent Workspace
PIW	Path Independent Workspace
RW	reaction wheel
RP	two-link arm with one revolute and one prismatic joint
RR, RRR	two-link, three-link arm with revolute joints
RBTC	Reduced Base-Torque Controller
SMM	Satellite Mounted Manipulator
SSA	Simultaneous (dynamic and CMG) Singularity Avoidance

Abstract

This work examines multiple control concepts for satellite-mounted manipulators (SMM). The primary focus is on base-actuated concepts, which eliminate singularity problems associated with free-floating SMMs. A new form of the equations of motion for an n -link SMM is developed using a quasi-coordinate form of Lagrange's Equation. Alternative free-floating SMM designs are presented which eliminate dynamic singularities, but still experience difficulties due to the unactuated base. A new generic SMM controller is developed as a framework for various control concepts with and without base actuation. Momentum-constrained Jacobians are shown to produce better SMM tracking than fixed-base Jacobians, even when base motion feedback is incorporated into the controller. A variation of the generic controller, termed the Reduced Base-Torque Controller (RBTC), is introduced and shown to reduce attitude control costs significantly while retaining the advantages of base control. The RBTC uses a task priority technique, assigning the first priority to end-effector control and secondary priority to maintaining a zero angular momentum state. Finally, the SMM dynamic model and generic controller are modified to include a cluster of control moment gyroscopes (CMG), and the effects of using the cluster for base attitude control in the SMM system are considered. A controller variation is developed which avoids singularities of both the CMG cluster and the manipulator system. A variety of planar and spatial simulations are used to validate the performance of the controllers. The results indicate that the base attitude control concept is the most viable SMM control concept in terms of tracking performance and singularity avoidance.

Singularity Avoidance Strategies For Satellite Mounted Manipulators Using Attitude Control

Chapter 1 - Introduction

1.1 Motivation

Since the beginning of man's exploration of space, an important role for robots has been evident. Given the hostile nature of the space environment, robots have often provided a safer and less expensive alternative to manned missions. For tasks too complex for autonomous robots, these devices have served as tools to augment natural human abilities. Examples of robot use in space range from surface rovers for planetary exploration to the Space Shuttle's Remote Manipulator System for a variety of tasks in orbit around Earth. As the fields of robotics and teleoperation mature, the role of robots in space will surely increase.

In recent years, a concept which has attracted significant interest is the satellite-mounted manipulator (SMM). Such a device could provide the flexibility required for many tasks while having a greater reach than current manned space operations. An SMM could be used for a variety of missions, including satellite maintenance, repair, and retrieval. The potential of SMMs is acknowledged by the National Aeronautics and Space Administration (NASA), as evidenced by several programs and studies. In the 1970's, the Flight Telerobotic Servicer was an early program to develop an SMM for on-orbit servicing. This program spawned several studies [8] [33], but was ultimately cancelled for budgetary reasons. Currently, NASA is working with the University of Maryland's Space Systems Laboratory on the RANGER program. RANGER is a satellite with a pair of seven degree-of-freedom (DOF) anthropomorphic manipulator arms, designed as a testbed for satellite mounted manipulators. It is due to launch in 1998 [1].

To date, most of the research on the control of space robots has centered on “free-floating” robots. Free-floating robots are defined as space robots where the satellite base position and orientation are unactuated. Without base actuation, no external forces or torques act upon the system when the end-effector is not in contact with the work environment. The linear and angular momenta of the system are conserved, so the spacecraft base moves in reaction to commanded arm motion. This interaction between arm and base has been noted in the operation of the Remote Manipulator System aboard NASA’s Space Shuttle [20]. The extent of the reaction depends upon the mass and inertia of the manipulator relative to the satellite base. If the arm represents a significant fraction of the combined system, the lack of base actuation can present problems for the robot controller. The conservation of angular momentum acts as a nonholonomic constraint on the system. This seriously complicates path planning, as the orientation of the base is dependent on the path of the arm trajectory in joint space.

Several authors have suggested control concepts for free-floating SMMs [2, 23, 30, 39, 44]. These concepts ensure that the manipulator’s objectives are met while compensating for, or even controlling, the base movement using only the manipulator’s joint actuators. Unfortunately, some of these methods fail in the neighborhood of singularities which can potentially occur throughout large regions of workspace. Other methods cannot precisely track a prescribed path between points in workspace.

The motivation for focussing on the free-floating mode of operation for SMMs is somewhat ambiguous in the literature. In one of the earliest papers on SMMs, Lindberg *et al.* [20] mention the poor interaction between the Shuttle attitude control system and the robot end-effector as the reason for disabling attitude control when the manipulator is operating. Other authors offer a range of motivating factors, from thruster exhaust contamination to actuator saturation and limited power

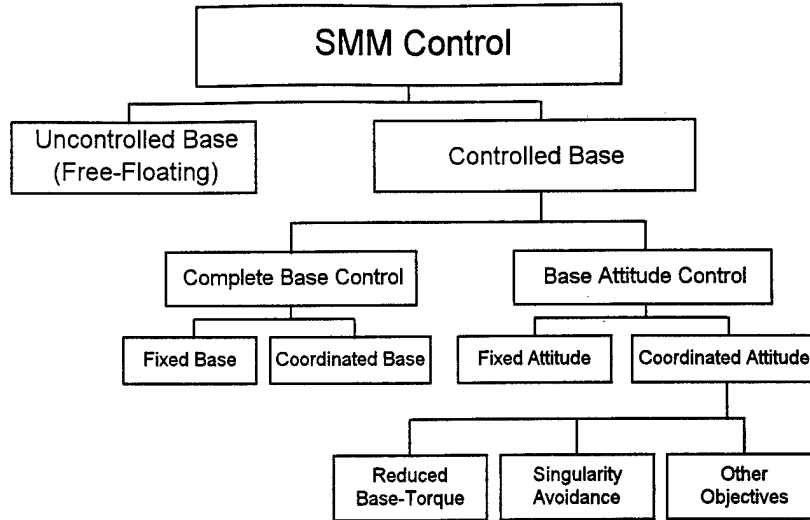


Figure 1. SMM Control Concept Hierarchy

storage capacity [31, 34, 46]. However, the most prevalent argument for the free-floater is the cost of base control in terms of thruster fuel [2, 30, 35, 44, 46].

Although all of these concerns have merit, the central thesis of this work is that base attitude control must play an important role in an SMM control system. Despite the disadvantages, base control offers substantial advantages. Controlling the base solves most singularity problems and greatly simplifies path planning by eliminating the path-dependent behavior of the system. It also creates redundancy which can be exploited in numerous ways.

Allowing the possibility of base control raises a number of questions. What is the goal of base control? Is it to fix the base and provide a stable platform so that the SMM performs like terrestrial robots, or should the base control be an integrated part of the manipulator system? Does this decision apply to both translation and attitude, or just attitude? What type of actuator is appropriate? And how does the actuator choice affect the controller goals? The options suggested by these questions form the basis for a hierarchy of SMM control concepts, as shown in Figure 1.

In this work, a preference for the controlled base concept is already clear from the earlier discussion. Referring to the concept hierarchy in Figure 1, the next consideration is whether to control the base in translation and attitude (Complete Base Control), or to control only base attitude, leaving the base free to translate in reaction to arm motion (Base Attitude Control). In making this choice, the benefits of translation control must be weighed against its cost. Base control is claimed to eliminate singularities, simplify path planning, and create redundancy. These benefits do not result equally from translation and attitude control. Translation control eliminates some singularities, but only those least likely to cause controller problems. Attitude control eliminates far more serious singularities which can be encountered throughout large areas of workspace. Since linear momentum is a holonomic constraint, translation control offers no advantage in path planning, while attitude control is essential to eliminating the path-dependent behavior of the system. The redundancy created by base control is evenly divided between translation and attitude control, each adding three additional controlled degrees of freedom. The disadvantage of translation control is the availability of actuators. Only thrusters can realistically be expected to provide the forces required to control the base translation during SMM operations. Thrusters require fuel, making them prohibitively expensive in this role. In addition, thrusters provide a much coarser level of control than joint motors, complicating efforts to achieve precise tracking. For these reasons, the added benefits of translation control are not worth the cost, making base attitude control the best alternative.

The next lower level of the concept diagram offers the option of fixed or coordinated control. This refers to the choice between using control to fix the base in workspace or to integrate base motion with the manipulator motion so as to enable the accomplishment of a secondary task. The advantage of fixed control is that if it is done well, the manipulator arm may be controlled using any control technique available to terrestrial robots. If the base is fixed in attitude only, some kinematic constants must be weighted by appropriate mass ratios, but it is essentially equivalent to a terrestrial

robot. There are several disadvantages to fixed control. It can require very high force/torque levels to maintain a stable platform. More importantly, the inherent redundancy of the base-controlled SMM is used completely in fixing the base, so that no secondary tasks are possible. By using coordinated control, the system is free to use the redundancy to optimize the control in some way.

If coordinated control is chosen, there are a number of secondary tasks that can be considered. The strategy adopted in this work is to use the redundancy to alleviate the costs of base control while retaining the various benefits. The cost of base control is directly related to the choice of actuator. The three most viable actuator choices for attitude control are: 1) Thrusters, 2) Control Moment Gyroscopes (CMG), and 3) Reaction Wheels (RW).

Thrusters represent the most straightforward option, but at the highest cost. Because they rely on external torques to control the base attitude, they are simple to incorporate into the dynamic model. However, thrusters use fuel which is a valuable nonrenewable resource for the system. It is generally best saved for tasks that cannot be performed without thrusters, such as orbit changing maneuvers. If thrusters must be used (perhaps to avoid redundant actuator systems), conserving fuel would be an important goal of the controller, second only to performing the commanded end-effector motion.

Control moment gyroscopes (CMGs) are perhaps the most attractive option for attitude control [24]. CMGs use renewable electrical power, making their cost in spacecraft resources similar to the cost of using the manipulator joints. In addition, they offer a large torque capability for their size [4]. The disadvantage of CMGs lies in their analytical complexity. The CMGs add significant nonlinear effects to the system dynamic model, due to the constant high rotational velocity of their rotors. Furthermore, a CMG cluster can experience singularity problems analogous to those of a manipulator arm. If CMGs are to be used for attitude actuation, avoiding cluster singularities is an obvious secondary task for the SMM controller.

Reaction wheels offer a sort of compromise between thrusters and CMGs. Like CMGs, they use electrical power. However, they do not suffer from singularity problems, offering a simple torque mechanism similar to thrusters. The primary disadvantage of reaction wheels is that they are significantly more massive than CMGs for the same torque capability [4]. Reaction wheels large enough to produce the torques needed for SMM operations may be prohibitively massive, so they are not considered in detail in this work. It may be noted, however, that if reaction wheels were used, they allow greater freedom in choosing a secondary task, since neither fuel nor singularities are an issue in their case.

1.2 Research Objectives

The major goal of this research is to develop and demonstrate viable base attitude concepts for satellite-mounted manipulators. In support of this goal, several research objectives are identified:

1. Explore alternative Free-Floating designs to eliminate singularity problems. (This objective is intended to ensure that no better alternatives to base attitude control can be found by simple extensions of singularity reduction methods used for terrestrial robots.)
2. Demonstrate how base control can eliminate singularities.
3. Develop an SMM controller using a workspace-based method that can follow a precise path.
4. Incorporate a CMG cluster into the SMM dynamic model and controller.
5. Explore the interaction of SMM and CMG cluster singularities.
6. Develop methods for reducing the costs of base attitude control.

1.3 Overview

These central ideas of the introduction are expanded upon in the body of this thesis. Chapter 2 provides an overview of the research to date applicable to the control of space robots. The kinematics and dynamics of SMMs are discussed in Chapter 3. The chapter reviews some basic theory of fixed-base robots and spacecraft dynamics. Topics include forward kinematics, manipulator Jacobians, the Lagrangian method of deriving equations of motion, and the use of quasicordinates to model spacecraft rotational dynamics. Chapter 4 introduces the Generalized Jacobian Matrix and dynamic singularities. Dynamic singularities are shown to depend only on the arm configuration and the inertial properties of the system. The effects of redundancy, prismatic joints, joint limits, and base control on dynamic singularities are investigated. Incorporating various combinations of these features leads to designs which eliminate or alleviate the problems caused by dynamic singularities. Chapter 5 investigates SMM control concepts. The advantages and disadvantages of full base control and base attitude control form the central focus of the chapter. A method for reduced base torque control using a task priority scheme is presented, and simulations show the superiority of the method over free-floating and earlier base control concepts. A Lyapunov controller is developed and demonstrated for the joint space portion of the control problem. Chapter 6 analyzes the use of control moment gyros as the base attitude control mechanism. The equations of motion for the system are revised to include multiple CMGs, and the effects of singularities of CMG clusters are considered. A method of avoiding CMG singularities and SMM dynamic singularities is constructed. Simulations demonstrate this controller for a three DOF arm mounted on a rigid satellite base with a three CMG cluster. Chapter 7 provides a summary, offers conclusions, and highlights the original contributions of this work.

Chapter 2 - Review of SMM-Related Research

Many researchers have investigated problems related to SMMs over the years. From the early analyses of multiple rigid body systems in the 1960's, through the technology and design studies of the Shuttle RMS in the 1970's and the emergence of space robotics as a sub-specialty of robotics in the 1990's, there is a rich body of literature from which to draw upon in the construction of a solution to the SMM control problem.

Much of the recent work in the space robotics field concentrates on solving the problem of controlling the end-effector of a manipulator on an uncontrolled, or "free-floating" base. The earliest works address the kinematics coupled with the linear and angular momentum conservation equations. Longman used the term "kinetics" to describe this motion [21]. In this paper, and in another by Lindberg, Longman and Zedd, the effects of the dynamic coupling of arm motion and base motion were first illustrated [19]. They developed a straightforward vector-based approach to obtain the forward and inverse kinematic solutions. They noted the path-dependent nature of the problem which results in non-unique solutions. An additional effect of the path dependence is the need in any solution for an integration over the path to determine the final inertial position. They also considered the kinematics when the base satellite has a fixed attitude, but is free to translate.

Umetani and Yoshida used conventional robotic kinematic relations to derive the end-effector velocity in terms of the base velocity and the joint velocities [43]. Using the two momentum conservation laws, they eliminated the base velocity to create the Generalized Jacobian Matrix (GJM) which maps joint velocities to end-effector velocity. The GJM is fundamental to much of the subsequent space robotics literature. In later papers, Yoshida and Umetani suggested a control method based on the resolved rate control algorithm [44], and explored workspace issues, introducing the

term *guaranteed workspace* for the reduced area of total workspace in which all trajectories are free of GJM singularities.

The third major work in kinematics of space robotics is the paper by Vafa and Dubowsky [45, 46]. They proposed a virtual manipulator which has joint angles and end-effector position identical to the real SMM, but has an inertially fixed base at the center of mass of the SMM system, and link lengths dependent on certain mass ratios. This approach allows direct application of earth-based robotics algorithms to the virtual manipulator to solve the SMM kinematics.

A recent paper by Saha [40] presented a generalized formulation of the Umetani and Yoshida approach. The paper expressed the kinematics and momentum equations in terms of an arbitrary "primary" body, and demonstrated that the choice of primary body can affect the numerical efficiency of a control algorithm. He concluded that for an end-effector tracking task, the end-effector should be the primary body if the base motion is not a concern.

Nakamura and Mukherjee were the first to explore fully the nonholonomic nature of the free-flying SMM [30]. They rigorously proved that the angular momentum conservation equation is a nonholonomic constraint. They constructed a nonlinear state space model of a six-DOF arm mounted on a free-flying base and proposed a path planning algorithm based on a Lyapunov function. In a later paper, they introduced the notion of nonholonomic redundancy [31]. This redundancy is evident by the ability of the SMM to reach the same inertial end-effector position using different joint angles.

Reyhanoglu and McClamroch provided a rigorous mathematical treatment of the planar case for free-flying systems. They proved that multi-body systems must have at least three bodies for controllability, but in any system with three or more links, all joint configurations are accessible. They also proved that no smooth feedback law will stabilize the system [38]. However, they presented an open-loop control law for the system using techniques from differential geometry [39].

Their controller first moves the planar robot to the required joint angles, and then uses cyclic motion to drive the base body to the desired orientation. Their proof is for free-floaters, and based upon formulating the problem completely in joint space. The paper does not consider how to translate this result to a workspace controller.

Papadopoulos and Dubowsky were the first to investigate singularities of SMM systems [35] [7]. They noted that the system is physically unable to move in some direction when the GJM becomes singular. Citing the dependence of the singularities on the inertial properties of the system, they described this type of singularity as a “dynamic singularity.” They showed that these dynamic singularities do not depend on the base position or orientation, and are eliminated when base attitude is fixed. They demonstrated that inertial workspace can be divided into regions where dynamic singularities are possible and not possible, introducing the terms Path Dependent Workspace (PDW) and Path Independent Workspace (PIW), respectively, to describe these regions.

The works cited above all concentrate on the kinematics of SMMs. Analysis of the dynamics of SMMs and specifically the generation of equations of motion (EOM) is less prevalent in the robotics literature. Yoshida and Umetani mentioned the problem in Ref. [51], where they wrote an expression for kinetic energy in terms of joint velocities (again using momentum conservation to eliminate base velocity) and then suggested that Lagrange’s equations will give the EOM. Luo and Sakawa [23] developed the same idea in more detail, using the result in a joint torque controller based on resolved acceleration control. Mukherjee and Nakamura briefly considered the EOM, describing a novel method using the Recursive Newton-Euler algorithm common for earth-based robots with the momentum equations incorporated as acceleration constraints [26].

Earlier papers, by dynamicists rather than roboticists, provided dynamic analyses of multiple rigid bodies that are applicable to SMMs. Hooker and Margulies [14] developed the first attitude equations for an n -body satellite using an Eulerian approach. Their only significant assumption

was that the bodies form an “open topological tree” (i.e. no closed loops). Their method eliminated constraint forces at the joints, but the EOM contain constraint torques resulting from less than three-DOF at the joints. Hooker [12] extended the method to include an arbitrary number (0-3) of rotational DOF at each joint, eliminating constraint torques from the EOM. Conway and Widhalm further modified the Hooker-Margulies equations by incorporating a translational degree of freedom at a joint [5]. In Hooker’s last paper [13], he abandoned most of this previous work in favor of an approach which uses a point on a specified “base body” rather than the system center of mass as the main translational point of reference for the system. This new method resulted in a somewhat simpler notation for the EOM. Other authors, Quinn [37], Hughes [15], Likins [18], and others provided many alternatives for generating EOM.

In contrast to the significant body of literature on free-flying space robots, the studies of controlled-base SMMs are few in number. There were, as mentioned above, some discussions of the kinematics of SMMs with fixed attitude in Refs. [20] and [45], but variable attitude control was not considered. However, some attempts to address this problem exist. Alexander and Cannon [2] developed a controller for the SMM that uses known information about the base controller forces and torques in the generation of commands for the arm joints. Oda [32] suggested the reverse philosophy, proposing a coordinated control architecture in which the effects of the arm motion are predicted and compensated for by the base attitude controller in a feedforward mode. If the current attitude is too far from a nominal position, the base attitude controller restricts the arm motion until the spacecraft attitude is back within a specified range. In both of these schemes, the satellite and the manipulator were viewed as distinct systems each with their own controllers. Each method was designed to handle undesirable controller interactions which can result from this control concept. Spofford and Akin [41] combined the manipulator and satellite base into a single system. They noted that using base control with arm control creates a redundant system. They demonstrated a

straightforward method of resolving this redundancy which tends to rely heavily on the base actuators to achieve the desired end-effector motion. They presented a method for lowering the use of base actuation by dynamically blending the straightforward method with another control scheme which uses only the arm joint actuators, but includes base velocity feedback to compensate for the base reactive motion. They weighted the proportion of each control scheme's contribution to the final control by means of a potential function.

Wee and Walker offered an analysis of the dynamics of an SMM integrated with a reaction wheel controlled base. In Refs. [48] and [49], they described the kinematics of the base by a massless virtual manipulator (not the same virtual manipulator as Vafa and Dubowsky [46]) which has three prismatic joints for base translation and three revolute joints for base orientation. Using a Lagrangian approach they derived equations of motion and showed that the revolute joints on the virtual manipulator are controllable by the reaction wheel (RW) torques. The RW torques are in the body frame while generalized torques associated with the virtual joints for orientation are in three different reference frames. This creates a need for some complicated transformations in the EOM, and introduces a coordinate-based singularity (similar to an Euler angle coordinate singularity) into the EOM that does not exist in the real system.

Chapter 3 - Kinematics & Dynamics of SMMs

3.1 Kinematics of Fixed-Base Manipulators

The kinematics of robots is generally divided into three main categories: forward kinematics, inverse kinematics, and velocity kinematics. Forward kinematics is the problem of determining the position of the end-effector (or other point of interest on the robot) in terms of the joint variables. It is a straightforward calculation, and the most significant part of any solution is the notation. A common approach is the use of a homogeneous transform for each DOF, using matrix multiplication as the rule of composition [27, 42]. Inverse kinematics is a much more difficult problem, and for some robots no closed form solution exists. When closed form solutions can be found for a particular robot geometry, the solution is often not unique. Because of the difficulties associated with the inverse kinematics, many analyses of robot kinematics are done at the velocity level. The fundamental tool is the *manipulator Jacobian*, which maps joint velocities to end-effector velocities. The Jacobian is derived from the forward kinematics and is well suited for both analysis and computation.

To construct the manipulator Jacobian, begin with the forward kinematic relation

$$r_e = f(\theta) \quad (1)$$

where r_e is the position vector of the end-effector in inertial space components and $f(\theta)$ is a non-linear function of the joint variables, $\theta \in \mathbb{R}^n$. Differentiating both sides with respect to time gives the velocity relation

$$\dot{r}_e = v_e = \frac{\partial f(\theta)}{\partial \theta} \dot{\theta} = J_p(\theta) \dot{\theta} \quad (2)$$

where J_p is a $3 \times n$ matrix that is a function of the manipulator's configuration. Both Eqs. (1) and (2) relate work space to joint space, but the Jacobian equation is linear with respect to the velocities.

As such, it is easily inverted,

$$\dot{\theta} = J_p^{-1}(\theta) v_e \quad (3)$$

to provide the necessary joint velocities for a desired end-effector velocity. The method of inversion can vary depending on the particular circumstances. In addition to the standard matrix inverse, there are a host of other inverses used in the robotics literature, the most common being the Moore-Penrose generalized inverse or pseudoinverse. When $n = 3$, the standard matrix inverse will serve as long as $J_p(\theta)$ is nonsingular. However, there are configurations of the robot that will be singular and many studies are devoted to methods of handling singularities. For robots where $n \neq 3$ there are additional concerns. When $n < 3$, there may not be any solutions, and when $n > 3$, there may be zero, one, or infinitely many solutions. In the latter case, the robot is said to be kinematically redundant, and many schemes exist for taking advantage of redundancy.

In this work, most of the examples assume that only end-effector position is of importance. Thus for planar cases, \dot{r}_e will be 2×1 , and for spatial cases, it will be 3×1 . However, in some developments where retaining the generality does not add unnecessary complexity to the equations, both the end-effector position and angular orientation are considered. In these instances, r_e is 6×1 , and the Jacobian definition is extended to include angular orientation in addition to position, so that Eq. (2) becomes

$$\begin{bmatrix} v_e \\ \omega_e \end{bmatrix} = \begin{bmatrix} J_p(\theta) \\ J_a(\theta) \end{bmatrix} \dot{\theta} = J_{fb}(\theta) \dot{\theta} \quad (4)$$

Efficient methods for constructing the manipulator Jacobians can be found in most robotics texts (see for example Refs. [29, 42]).

3.2 Kinematics of an n -Link SMM

For satellite-mounted manipulators, the problem of describing the system kinematics is quite similar to the fixed-base manipulator. The major difference is that the forward kinematics now must include the position and orientation of the base as well as the manipulator configuration (joint

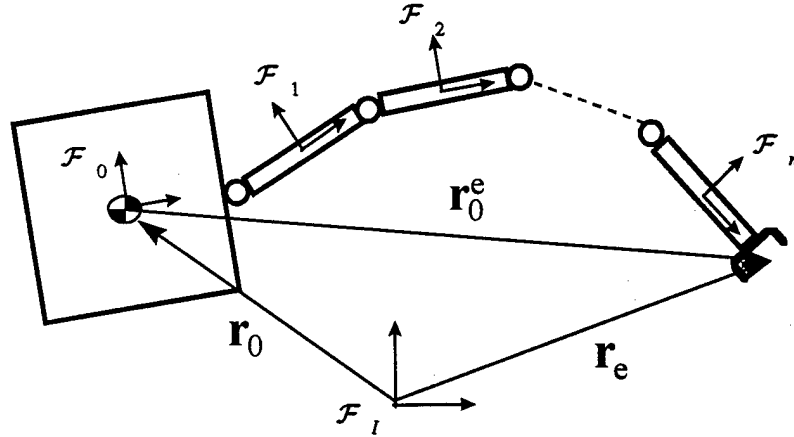


Figure 2. Satellite Mounted Manipulator (SMM)

angles). This makes solving the inverse kinematics problem more difficult, prompting our use of the velocity kinematics relations.

In the next chapter, the system momenta will be shown to play an important role in the traditionally kinematic problem of relating end-effector motion and joint motion. This has led to a number of different formulations for the kinematics, with some authors writing the equations in reference to the system center of mass and others writing them with respect to the spacecraft base. Saha [40] developed general equations which describe the system with respect to an arbitrary primary body, which can be chosen depending on the expected task. Since controlling the base motion will often be an important secondary consideration to controlling the end-effector, a formulation which directly includes base coordinates is used here.

Consider an SMM, shown in Figure 2, modeled as $n + 1$ rigid bodies connected in series by n revolute or prismatic joints. Reference frames are established as shown in the figure. \mathcal{F}_I is the inertial frame, established arbitrarily. \mathcal{F}_0 is fixed to the satellite (body-0) at its center of mass and aligned in the principal directions of the body. Each frame, \mathcal{F}_i , $i = 1, 2, \dots, n$, is fixed to body i with the choice of origin and orientation depending on the particular arm design, according to the

Denavit-Hartenberg convention [6]. The position of the end-effector \mathbf{r}_e can be described by the forward kinematics equation

$$\mathbf{r}_e = f(\mathbf{r}_0, \Omega, \theta) \quad (5)$$

where \mathbf{r}_0 is the inertial position of the satellite base's center of mass¹, Ω is the base orientation (the orientation of \mathcal{F}_0 with respect to \mathcal{F}_I), and θ is the set of arm joint angles. From Figure 2, one can write the vector expression

$$\mathbf{r}_e = \mathbf{r}_0 + \mathbf{r}_0^e \quad (6)$$

where \mathbf{r}_0^e is the position of the end-effector relative to the base center of mass. The inertial time derivative of Eq. (6) provides the corresponding velocity relation,

$$\dot{\mathbf{r}}_e = \dot{\mathbf{r}}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^e + \frac{{}^0d}{dt}(\mathbf{r}_0^e) \quad (7)$$

where $\boldsymbol{\omega}_0$ is the base angular velocity and ${}^0d/dt(\cdot)$ indicates differentiation with respect to frame \mathcal{F}_0 , rather than the inertial frame, \mathcal{F}_I . This equation can be written in matrix form as

$$\dot{\mathbf{r}}_e = J_{pv}v_0 + J_{p\omega}\boldsymbol{\omega}_0 + J_{p\theta}\dot{\theta} \quad (8)$$

where $\dot{\mathbf{r}}_e$ is the inertial velocity of the end-effector in \mathcal{F}_I components, v_0 is the inertial velocity of the base in \mathcal{F}_I components, $\boldsymbol{\omega}_0$ is the angular velocity of the base in \mathcal{F}_0 components and $\dot{\theta}$ are the joint velocities. The Jacobians, J_{pv} , $J_{p\omega}$ and $J_{p\theta}$ are

$$\begin{aligned} J_{pv} &= U_{3 \times 3} \\ J_{p\omega} &= -R_I^0 r_0^{e \times} \\ J_{p\theta} &= R_I^0 J_p \end{aligned} \quad (9)$$

where $U_{3 \times 3}$ is the 3×3 identity matrix, R_I^0 is the rotation matrix from the \mathcal{F}_I frame to the \mathcal{F}_0 frame, $r_0^{e \times}$ is the skew-symmetric matrix formed from the end-effector position relative to the base in

¹Throughout this work, the use of bold variables is restricted to vectors or dyadics where no specific component frame is implied in the equation. In equations where vectors or matrices are in component form, they are not bold and the reference frame is indicated with the first usage of the variable.

\mathcal{F}_0 components, and J_p is the kinematic manipulator Jacobian for a fixed-base arm, mapping joint velocity to end-effector velocity in the \mathcal{F}_0 frame.

In the most general case, r_e would include the orientation of the end-effector as well as position.

Then the end-effector velocity can be given by

$$\dot{r}_e = \begin{bmatrix} v_e \\ \omega_e \end{bmatrix} \quad (10)$$

where v_e is the translational velocity of the end-effector relative to the inertial frame, expressed in the inertial frame, but ω_e is the angular velocity of the end-effector relative to the inertial frame *expressed in the spacecraft body frame*. Equation (8) becomes

$$\dot{r}_e = J_v v_0 + J_\omega \omega_0 + J_\theta \dot{\theta} \quad (11)$$

$$= \begin{bmatrix} J_{pv} \\ J_{av} \end{bmatrix} v_0 + \begin{bmatrix} J_{p\omega} \\ J_{a\omega} \end{bmatrix} \omega_0 + \begin{bmatrix} J_{p\theta} \\ J_{a\theta} \end{bmatrix} \dot{\theta} \quad (12)$$

where J_{pv} , $J_{p\omega}$, and $J_{p\theta}$ are defined in Eq. (9), and J_{av} , $J_{a\omega}$, and $J_{a\theta}$ are given by

$$J_{av} = 0_{3 \times 3} \quad (13)$$

$$J_{a\omega} = U_{3 \times 3} \quad (14)$$

$$J_{a\theta} = J_a \quad (15)$$

In later chapters, Eq. (11) is occasionally used interchangeably with Eq. (8). The meaning should be clear from the definition of the end-effector velocity \dot{r}_e given in the particular case.

3.3 Equations of Motion of an Open Chain Fixed-Base Manipulator

For a dynamic system with n degrees of freedom (DOF), the motion of the system can be determined by the n differential equations known as Lagrange's Equations,

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_k} \right) - \frac{\partial L}{\partial q_k} = Q_k \quad (k = 1, 2, \dots, n) \quad (16)$$

where L is the quantity known as the *Lagrangian*. The Lagrangian is the difference of the kinetic and potential energies of the system, $L = T - V$. The q_k are known as generalized coordinates, and can be any set of coordinates that uniquely determine the state of the dynamic system. The Q_k

are known as generalized forces, and represent forces applied to the system in the “direction” of q_k . This method of determining the equations of motion is a commonly used method in the robotics literature [27, 42].

A typical robotic arm is comprised of a series of links connected by actuated joints in an open chain (i.e., there are no closed loops of links). For this type of robot, known as an open chain manipulator, the joint angles, θ_k , are a natural choice for the generalized coordinates. To determine Lagrange’s equations of motion for the robot, one begins by writing the Lagrangian of the manipulator in terms of the generalized coordinates, θ_k . The kinetic energy may be written in matrix form as

$$T = \frac{1}{2} \sum_{k=1}^n \left\{ m_k v_k^\top v_k + \omega_k^\top I_k \omega_k \right\} \quad (17)$$

where link k has mass m_k , center of mass velocity v_k , angular velocity ω_k , and body-fixed inertia I_k . A Jacobian, J_k , may be defined for link k which relates the velocity of link k to joint velocities in much the same way as the manipulator Jacobian relates end-effector velocity to joint velocities (see Eq. (4)). The defining equation is

$$\begin{bmatrix} v_k \\ \omega_k \end{bmatrix} \triangleq J_k(\theta) \dot{\theta} = \begin{bmatrix} J_{pk} \\ J_{ak} \end{bmatrix} \dot{\theta} \quad (18)$$

Substituting Eq. (18) into Eq. (17), the total kinetic energy of the robot is given by

$$T = \frac{1}{2} \sum_{k=1}^n \left\{ m_k \dot{\theta}^\top J_{pk}^\top J_{pk} \dot{\theta} + \dot{\theta}^\top J_{ak}^\top I_k J_{ak} \dot{\theta} \right\} \quad (19)$$

Now $\dot{\theta}$ can be brought outside the summation giving

$$T = \frac{1}{2} \dot{\theta}^\top M(\theta) \dot{\theta} \quad (20)$$

where

$$M(\theta) \triangleq \frac{1}{2} \sum_{k=1}^n \left\{ m_k J_{pk}^\top J_{pk} + J_{ak}^\top I_k J_{ak} \right\} \quad (21)$$

The matrix $M(\theta)$ is known as the manipulator inertia matrix. For terrestrial robots, a potential energy term is generally included in the Lagrangian to account for gravitational effects. However,

for an SMM with rigid links this term can be neglected², so the Lagrangian, L , is simply the kinetic energy, T , as given in Eq. (20).

Before applying Lagrange's Equation, Eq. (16), it is convenient to switch to index notation (see Appendix A for a review of this notation). The Lagrangian becomes

$$L = \frac{1}{2} M_{ij}(\theta) \dot{\theta}_i \dot{\theta}_j \quad (22)$$

Then the Lagrangian in Eq. (22) may be substituted into Eq. (16). Performing the differentiation gives the first term

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_i} \right) = \frac{d}{dt} (M_{ij} \dot{\theta}_j) = \dot{M}_{ij} \dot{\theta}_j + M_{ij} \ddot{\theta}_j \quad (23)$$

and the second term is

$$\frac{\partial L}{\partial \theta_i} = \frac{1}{2} \frac{\partial M_{kj}}{\partial \theta_i} \dot{\theta}_k \dot{\theta}_j \quad (24)$$

The time derivative of the manipulator inertia matrix, \dot{M}_{ij} , can be written as

$$\dot{M}_{ij}(\theta) = \frac{\partial M_{ij}}{\partial \theta_k} \dot{\theta}_k \quad (25)$$

Substituting Eqs. (23)-(25) into Eq. (16) gives

$$M_{ij} \ddot{\theta}_j + \frac{\partial M_{ij}}{\partial \theta_k} \dot{\theta}_k \dot{\theta}_j - \frac{1}{2} \frac{\partial M_{kj}}{\partial \theta_i} \dot{\theta}_k \dot{\theta}_j = Q_i \quad (26)$$

The second two terms can be combined, giving

$$M_{ij} \ddot{\theta}_j + \Gamma_{ijk} \dot{\theta}_j \dot{\theta}_k = Q_i \quad (27)$$

where

$$\Gamma_{ijk} = \frac{1}{2} \left(\frac{\partial M_{ij}}{\partial \theta_k} + \frac{\partial M_{ik}}{\partial \theta_j} - \frac{\partial M_{jk}}{\partial \theta_i} \right) \quad (28)$$

The functions, Γ_{ijk} , are known as the Christoffel symbols corresponding to the inertia matrix, M_{ij} . Note that Γ cannot be written in matrix form, prompting the use of index notation. To revert to matrix form, it is customary to define a $C(\theta, \dot{\theta})$ matrix by

$$C_{ij}(\theta, \dot{\theta}) = \Gamma_{ijk} \dot{\theta}_k \quad (29)$$

²The gravitational effects are negligible for the SMM because the entire system is in free-fall (orbit) around the earth, providing a so called "zero-gravity" environment. The gravity-gradient effect, which is often included in the spacecraft attitude analyses, is also negligible on the expected time scale of SMM operations.

This definition of C is not unique, but is the most common choice due to the compact form of representation with the Christoffel symbols and some other useful properties which appear in stability proofs. Unlike the inertia matrix, the C matrix is not symmetric in general.

Using the M and C matrices as defined, the equations of motion for a fixed-base manipulator can be written

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} = \tau \quad (30)$$

where Q is replaced by τ , the usual notation for the joint torques.

3.4 Lagrange's Equations Using Quasi-Coordinates

Lagrange's equations are written in terms of generalized coordinates q_k which are assumed to be true coordinates in the sense that if the velocities \dot{q}_k are known functions of time, they can be integrated to give the coordinates q_k . For spacecraft, equations of motion are often written in terms of body angular velocities since the inertia matrix for the spacecraft is constant in the body frame. Unfortunately, these angular velocities cannot be integrated to obtain true orientation coordinates of the spacecraft. Therefore, they cannot be used in the normal form of Lagrange's equation. However, the "quasi-coordinate" form of Lagrange's equation extends the powerful Lagrangian method, allowing it to handle this problem. The derivation below is done in index notation, but generally follows the one found in Meirovitch [25].

First, the angular velocity is written as a linear combination of the derivatives of a set of true coordinates

$$\omega_i = A_{ij}\dot{q}_j \quad (31)$$

where $A = A(q)$. A common choice for q is Euler angles³, but this method is not restricted to this representation. When the matrix A is nonsingular, this relation can be inverted to give

$$\dot{q}_i = B_{ij}\omega_j \quad (32)$$

³A discussion of Euler angles, as well as alternative orientation parameters, can be found in Ref. [15].

where B_{ij} is determined from the equation

$$B_{ij}A_{jk} = \delta_{ik} \quad (33)$$

where δ_{ik} is the Kronecker delta (see Appendix A)

The kinetic energy, $T = T(q, \dot{q})$, can then be written in terms of the angular velocity, ω . This new function is denoted as $\bar{T} = \bar{T}(q, \omega(q, \dot{q}))$. This expression replaces the former expression for kinetic energy in Lagrange's equation (16). The first term becomes

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_k} \right) = \frac{d}{dt} \left(\frac{\partial \bar{T}(q, \omega(q, \dot{q}))}{\partial \dot{q}_k} \right) = \frac{d}{dt} \left(\frac{\partial \bar{T}}{\partial \omega_i} \frac{\partial \omega_i}{\partial \dot{q}_k} \right) = \frac{d}{dt} \left(\frac{\partial \bar{T}}{\partial \omega_i} A_{ik} \right) \quad (34)$$

$$\frac{d}{dt} \left(\frac{\partial \bar{T}}{\partial \omega_i} A_{ik} \right) = \frac{d}{dt} \left(\frac{\partial \bar{T}}{\partial \omega_i} \right) A_{ik} + \frac{\partial \bar{T}}{\partial \omega_i} \frac{d}{dt} (A_{ik}) \quad (35)$$

since A is a function only of the coordinates q , its time derivative can be expressed by

$$\frac{d}{dt} (A_{ik}) = \frac{\partial A_{ik}}{\partial q_j} \dot{q}_j = \frac{\partial A_{ik}}{\partial q_j} B_{jl} \omega_l \quad (36)$$

The second term in Lagrange's equation becomes

$$\frac{\partial T}{\partial q_k} = \frac{\partial \bar{T}(q, \omega)}{\partial q_k} + \frac{\partial \bar{T}(q, \omega)}{\partial \omega_i} \frac{\partial \omega_i}{\partial q_k} = \frac{\partial \bar{T}}{\partial q_k} + \frac{\partial \bar{T}}{\partial \omega_i} \frac{\partial A_{ij}}{\partial q_k} B_{jl} \omega_l \quad (37)$$

Combining the terms and multiplying both sides by B_{km} gives the quasi-coordinate form of Lagrange's equation,

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial \bar{T}}{\partial \omega_m} \right) + \frac{\partial \bar{T}}{\partial \omega_i} \frac{\partial A_{ik}}{\partial q_j} B_{jl} \omega_l B_{km} - B_{km} \frac{\partial \bar{T}}{\partial q_k} \\ - \frac{\partial \bar{T}}{\partial \omega_i} \frac{\partial A_{ij}}{\partial q_k} B_{jl} \omega_l B_{km} = B_{km} Q_k \end{aligned} \quad (38)$$

After much algebra, the expression becomes

$$\frac{d}{dt} \left(\frac{\partial \bar{T}}{\partial \omega_m} \right) + \omega_i \frac{\partial \bar{T}}{\partial \omega_j} e_{ijm} - B_{km} \frac{\partial \bar{T}}{\partial q_k} = B_{km} Q_k \quad (39)$$

where e_{ijm} is the alternator function, or unit completely skew-symmetric third-order tensor, as defined in Appendix A. Now writing Eq. (39) back in matrix notation, this expression is

$$\frac{d}{dt} \left(\frac{\partial \bar{T}}{\partial \omega} \right) + \omega^\times \frac{\partial \bar{T}}{\partial \omega} - B^\top \frac{\partial \bar{T}}{\partial q} = B^\top Q \quad (40)$$

When the kinetic energy does not depend on the generalized coordinates, then $\partial \bar{T} / \partial q = 0$, and the expression can be further simplified to

$$\frac{d}{dt} \left(\frac{\partial \bar{T}}{\partial \omega} \right) + \omega^\times \frac{\partial \bar{T}}{\partial \omega} = B^\top Q \quad (41)$$

When $\bar{T} = \frac{1}{2} \omega^\top I \omega$, this translates directly to the familiar Euler equation for rigid bodies,

$$I \dot{\omega} + \omega^\times I \omega = M \quad (42)$$

However, when $\partial \bar{T} / \partial q_k \neq 0$, Eq. (39) can quickly become cumbersome in practice. Quinn [37] noted a useful relation for dealing with this term as it arises in rotating space structures. In these problems, the dependence of kinetic energy on orientation enters only via the rotation matrix between the inertial frame and the body-fixed frame. These terms in the expression for \bar{T} (denoted here by \bar{T}^*) are often of the form

$$\bar{T}^* = a^\top R^\top b \quad (43)$$

where a is a vector in the body-fixed frame, b is a vector in the inertial frame, and R is the rotation from the body-fixed to the inertial frame. Then the last term in Eq. (39) can be written

$$\begin{aligned} B_{km} \frac{\partial \bar{T}}{\partial q_k} &= B_{km} a_i \frac{\partial (R^\top)_{ij}}{\partial q_k} b_j \\ &= a_i e_{lim} (R^\top)_{ij} b_j \end{aligned} \quad (44)$$

or in matrix notation,

$$B^\top \frac{\partial \bar{T}}{\partial q} = a^\times R^\top b \quad (45)$$

This equation, used with Eq. (39), provides a systematic method of determining the equations of motion associated with the unconstrained rotation of a body in space. The equations associated with true coordinates are still found with the original form of Lagrange's equation, and are unaffected by the choice of representation for kinetic energy ($T(q, \dot{q})$ or $\bar{T}(q, \omega)$).

3.5 Equations of Motion of an n -Link SMM

The equations of motion for an n -link SMM can be generated by the Lagrangian approach, using the quasi-coordinate form for the equations associated with the base angular motion. The goal is to obtain equations in a form similar to the equations of motion for the open-chain manipulator, Eq. (30), developed in Section 3.3. The derivation below draws on elements of that section as well as portions of the discussion of quasi-coordinates in Section 3.4.

The first step is to write the Lagrangian, which for the n -link SMM with rigid links is the total kinetic energy of the system. The total kinetic energy can be expressed as the sum of the kinetic energy of each body in the system,

$$T = \sum_{i=0}^n T_i \quad (46)$$

The kinetic energy of body- i is

$$T_i = \frac{1}{2} m_i \mathbf{v}_i \cdot \mathbf{v}_i + \frac{1}{2} \boldsymbol{\omega}_i \cdot \mathbf{I}_i \cdot \boldsymbol{\omega}_i \quad (47)$$

where \mathbf{v}_i is the velocity of the body's center of mass, $\boldsymbol{\omega}_i$ is the body's angular velocity, m_i is the body's mass, and \mathbf{I}_i is the body's inertia tensor. Denoting the linear and angular velocity of the satellite base by \mathbf{v}_0 and $\boldsymbol{\omega}_0$, respectively, expanded expressions for the velocities of the links may be written,

$$\mathbf{v}_i = \mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^i + \mathbf{v}_0^i \quad (48)$$

$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_0 + \boldsymbol{\omega}_0^i \quad (49)$$

where \mathbf{r}_0^i , \mathbf{v}_0^i , and $\boldsymbol{\omega}_0^i$ are the quantities associated with link i with respect to the satellite base. The relative position vector, \mathbf{r}_0^i , is shown in Figure 3.

Substituting Eqs. (47)-(49) into Eq. (46), and expanding, the kinetic energy is

$$\begin{aligned} T = & \frac{1}{2} m_0 \mathbf{v}_0 \cdot \mathbf{v}_0 + \frac{1}{2} \boldsymbol{\omega}_0 \cdot \mathbf{I}_0 \cdot \boldsymbol{\omega}_0 \\ & + \sum_{i=1}^n \left\{ \frac{1}{2} m_i (\mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^i + \mathbf{v}_0^i) \cdot (\mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^i + \mathbf{v}_0^i) \right. \\ & \left. + \frac{1}{2} (\boldsymbol{\omega}_0 + \boldsymbol{\omega}_0^i) \cdot \mathbf{I}_i \cdot (\boldsymbol{\omega}_0 + \boldsymbol{\omega}_0^i) \right\} \end{aligned} \quad (50)$$

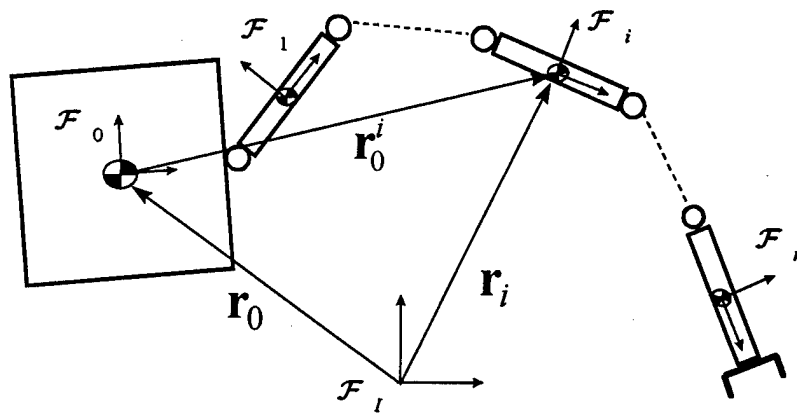


Figure 3. Defining the position of a link relative to the base center of mass.

Expanding the terms in the summation and writing in matrix form, this expression becomes

$$\begin{aligned}
T = & \frac{1}{2}m_0v_0^\top v_0 + \frac{1}{2}\omega_0^\top I_0\omega_0 + \frac{1}{2}\sum_{i=1}^n \left\{ m_i \left[v_0^\top v_0 + v_0^{i\top} v_0^i \right. \right. \\
& - \omega_0^\top r_0^{i\times} r_0^{i\times} \omega_0 + 2v_0^\top R_I^0 v_0^i \\
& \left. \left. - 2v_0^\top R_I^0 r_0^{i\times} \omega_0 + 2\omega_0^\top r_0^{i\times} v_0^i \right] \right. \\
& \left. + \omega_0^\top R_0^i I_i R_0^{i\top} \omega_0 + \omega_0^{i\top} R_0^i I_i R_0^{i\top} \omega_0^i + 2\omega_0^\top R_0^i I_i R_0^{i\top} \omega_0^i \right\}
\end{aligned} \tag{51}$$

where v_0 is in \mathcal{F}_I components, ω_0 , ω_0^i , r_0^i , and v_0^i are in \mathcal{F}_0 components, and the I_i are in \mathcal{F}_i components. The link velocities relative to the base, v_0^i and ω_0^i , may be written in terms of the joint velocities using Jacobians as in Section 3.3. That is,

$$v_0^i \triangleq J_{pi}(\theta)\dot{\theta} \tag{52}$$

$$\omega_0^i \triangleq J_{ai}(\theta)\dot{\theta}$$

Substituting Eq. (52) into Eq. (51), gives

$$\begin{aligned}
T = & \frac{1}{2}m_0v_0^\top v_0 + \frac{1}{2}\omega_0^\top I_0\omega_0 + \frac{1}{2}\sum_{i=1}^n \left\{ m_i \left[v_0^\top v_0 + \dot{\theta}^\top J_{pi}^\top J_{pi} \dot{\theta} \right. \right. \\
& - \omega_0^\top r_0^{i\times} r_0^{i\times} \omega_0 + 2v_0^\top R_I^0 J_{pi} \dot{\theta} \\
& \left. \left. - 2v_0^\top R_I^0 r_0^{i\times} \omega_0 + 2\omega_0^\top r_0^{i\times} J_{pi} \dot{\theta} \right] \right. \\
& \left. + \omega_0^\top R_0^i I_i R_0^{i\top} \omega_0 + \dot{\theta}^\top J_{ai}^\top R_0^i I_i R_0^{i\top} J_{ai} \dot{\theta} + 2\omega_0^\top R_0^i I_i R_0^{i\top} J_{ai} \dot{\theta} \right\}
\end{aligned} \tag{53}$$

Defining a column matrix of the generalized velocities,

$$\dot{q} \triangleq \begin{bmatrix} v_0 \\ \omega_0 \\ \dot{\theta} \end{bmatrix} \tag{54}$$

the kinetic energy can be written

$$T = \frac{1}{2}\dot{q}^\top M(q)\dot{q} \tag{55}$$

where

$$M(q) = \begin{bmatrix} M_v & M_{v\omega} & M_{v\theta} \\ M_{v\omega}^\top & M_\omega & M_{\omega\theta} \\ M_{v\theta}^\top & M_{\omega\theta}^\top & M_\theta \end{bmatrix} \tag{56}$$

$$M_v = m_0 U_{3 \times 3} + \sum_{i=1}^n m_i U_{3 \times 3} \quad (57)$$

$$M_\omega = I_0 + \sum_{i=1}^n \left\{ R_0^i I_i R_0^{i\top} - m_i r_0^{i \times} r_0^{i \times} \right\} \quad (58)$$

$$M_\theta = \sum_{i=1}^n \left\{ m_i J_{pi}^\top J_{pi} + J_{ai}^\top R_0^i I_i R_0^{i\top} J_{ai} \right\} \quad (59)$$

$$M_{v\omega} = - \sum_{i=1}^n m_i R_I^0 r_0^{i \times} \quad (60)$$

$$M_{v\theta} = \sum_{i=1}^n m_i R_I^0 J_{pi} \quad (61)$$

$$M_{\omega\theta} = \sum_{i=1}^n \left\{ m_i r_0^{i \times} J_{pi} + R_0^i I_i R_0^{i\top} J_{ai} \right\} \quad (62)$$

Using this expression for the kinetic energy of the system, Lagrange's equation can be applied to obtain the equations of motion. As stated earlier, the goal is to put the equations of motion into the form

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} = Q \quad (63)$$

Where $M(q)$ is the inertia matrix defined above, $C(q, \dot{q})$ is the matrix containing all of the Coriolis and centrifugal terms, and Q is a vector of the generalized forces.

In the development of the open chain manipulator EOM, the application of Lagrange's equation led to an expression of the C matrix as a function of the inertia matrix and its derivatives. This function was compactly described using Christoffel symbols. This compact representation is not possible for the n -link SMM because the quasi-coordinate form of Lagrange's equation must be used to derive some of the EOM. While the terms represented by the product, $C(q, \dot{q})\dot{q}$, are unique,

the definition of the C matrix is not. In the remainder of this section, one convenient representation of the C matrix is developed.

First, it is useful to note that the kinetic energy can be written in terms of the inertia submatrices defined in Eqs. (57)-(62) as

$$\begin{aligned} T = \frac{1}{2} \{ & v_0^\top M_v v_0 + v_0^\top M_{v\omega} \omega_0 + v_0^\top M_{v\theta} \dot{\theta} \\ & + \omega_0^\top M_{\omega v} v_0 + \omega_0^\top M_{\omega\omega} \omega_0 + \omega_0^\top M_{\omega\theta} \dot{\theta} \\ & + \dot{\theta}^\top M_{\theta v} v_0 + \dot{\theta}^\top M_{\theta\omega} \omega_0 + \dot{\theta}^\top M_{\theta\theta} \dot{\theta} \} \end{aligned} \quad (64)$$

It is convenient to partition the C matrix in the same manner as the inertia matrix,

$$C(q, \dot{q}) = \begin{bmatrix} C_v & C_{v\omega} & C_{v\theta} \\ C_{\omega v} & C_{\omega\omega} & C_{\omega\theta} \\ C_{\theta v} & C_{\theta\omega} & C_{\theta\theta} \end{bmatrix} \quad (65)$$

Recall that, unlike the inertia matrix, the C matrix is not generally symmetric.

For the base translation equations, apply the standard form of Lagrange's equation (recall that there are no potential energy terms, so $L = T$),

$$\frac{d}{dt} \left(\frac{\partial T}{\partial v_0} \right) - \frac{\partial T}{\partial r_0} = Q_r \quad (66)$$

The second term on the left hand side is zero, as an examination of the inertia matrix definition (Eqs.

(57)-(62)) reveals no dependence on the base position. The first term can be expanded as

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial T}{\partial v_0} \right) &= \frac{d}{dt} (M_v v_0 + M_{v\omega} \omega_0 + M_{v\theta} \dot{\theta}) \\ &= \dot{M}_v v_0 + M_{v\omega} \dot{\omega}_0 + M_{v\theta} \ddot{\theta} + \dot{M}_v v_0 + \dot{M}_{v\omega} \omega_0 + \dot{M}_{v\theta} \dot{\theta} \end{aligned} \quad (67)$$

The first three terms are incorporated into the first term of Eq. (63), while the second three terms are incorporated into the second term (the C matrix term). It is apparent from the equation above that a convenient way to form the C matrix (recall that it is not the only way) is to define

$$\begin{aligned} C_v &= \dot{M}_v \\ C_{v\omega} &= \dot{M}_{v\omega} \end{aligned} \quad (68)$$

$$C_{v\theta} = \dot{M}_{v\theta}$$

The inertia matrix derivatives can be found by differentiating Eqs. (57), (60), and (61),

$$\dot{M}_v = \frac{d}{dt} (m_T U_3) = 0 \quad (69)$$

$$\begin{aligned} \dot{M}_{v\omega} &= \frac{d}{dt} \left(- \sum_{i=1}^n \{ m_i R_I^0 r_0^{i \times} \} \right) \\ &= - \frac{d}{dt} (R_I^0) \sum_{i=1}^n \{ m_i r_0^{i \times} \} \end{aligned} \quad (70)$$

$$\begin{aligned} &- R_I^0 \frac{d}{dt} \left(\sum_{i=1}^n \{ m_i r_0^{i \times} \} \right) \\ &= - R_I^0 \omega_0^\times \sum_{i=1}^n \{ m_i r_0^{i \times} \} - R_I^0 \sum_{i=1}^n \left\{ m_i (J_{pi} \dot{\theta})^\times \right\} \\ \dot{M}_{v\theta} &= \frac{d}{dt} \left(\sum_{i=1}^n m_i R_I^0 J_{pi} \right) \\ &= R_I^0 \omega_0^\times \sum_{i=1}^n m_i J_{pi} + R_I^0 \sum_{i=1}^n \left\{ m_i \frac{\partial J_{pi}}{\partial \theta} \right\} \dot{\theta} \\ &= (R_I^0 \omega_0)^\times M_{v\theta} + \frac{\partial M_{v\theta}}{\partial \theta} \dot{\theta} \end{aligned} \quad (71)$$

In the above equations, the relation $\dot{R}_I^0 = R_I^0 \omega_0^\times$ has been used. This identity is derived in Appendix B.1. Also, the notation in the $\dot{M}_{v\theta}$ equation is somewhat ambiguous, since partial derivatives of matrices with respect to vectors are not matrices, but 3rd order tensors. In this equation, and in subsequent equations where this notation is used, the intended meaning is defined in terms of index notation as

$$\left(\frac{\partial M_{v\theta}}{\partial \theta} \dot{\theta} \right)_{ij} \triangleq \frac{\partial (M_{v\theta})_{ij}}{\partial \theta_k} \dot{\theta}_k \quad (72)$$

In practice, taking partial derivatives of inertia matrix partitions with respect to θ is quite tedious, and can be delegated to symbolic software such as *Mathematica* for reasonably sized systems.

Next, the EOM associated with ω_0 can be determined using the quasi-coordinate form of Lagrange's equation,

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \omega_0} \right) + \omega_0^\times \left(\frac{\partial T}{\partial \omega_0} \right) - \frac{\partial T}{\partial R_I^{0T}} = Q_\omega \quad (73)$$

The first term is

$$\begin{aligned}\frac{d}{dt} \left(\frac{\partial T}{\partial \omega_0} \right) &= \frac{d}{dt} (M_{\omega v} v_0 + M_{\omega} \omega_0 + M_{\omega \theta} \dot{\theta}) \\ &= \dot{M}_{\omega v} v_0 + M_{\omega} \dot{\omega}_0 + M_{\omega \theta} \ddot{\theta} + \dot{M}_{\omega v} v_0 + \dot{M}_{\omega} \omega_0 + \dot{M}_{\omega \theta} \dot{\theta}\end{aligned}\quad (74)$$

The second term is

$$\omega_0^\times \left(\frac{\partial T}{\partial \omega_0} \right) = \omega_0^\times (M_{\omega v} v_0 + M_{\omega} \omega_0 + M_{\omega \theta} \dot{\theta}) \quad (75)$$

and the third term is

$$-\frac{\partial T}{\partial R_I^{0\top}} = -\frac{\partial}{\partial R_I^{0\top}} (\omega_0^\top M_{\omega v} v_0 + \dot{\theta}^\top M_{\theta v} v_0) \quad (76)$$

Using the relation in Section 3.3.4, it can be shown that Eq. (76) becomes

$$\begin{aligned}\frac{\partial T}{\partial R_I^{0\top}} &= \left(\omega_0^\top \sum_{i=1}^n m_i r_0^{i \times} \right)^\times R_I^{0\top} v_0 \\ &\quad + \left(\dot{\theta}^\top \sum_{i=1}^n \{m_i J_{pi}^\top\} \right)^\times R_I^{0\top} v_0\end{aligned}\quad (77)$$

Then the ω_0 row of the C matrix may be defined by

$$\begin{aligned}C_{\omega v} &= \dot{M}_{\omega v} + \omega_0^\times M_{\omega v} \\ &\quad - \left(\sum_{i=1}^n m_i J_{pi} \dot{\theta} - \sum_{i=1}^n m_i r_0^{i \times} \omega_0 \right)^\times R_I^{0\top} \\ C_{\omega} &= \dot{M}_{\omega} + \omega_0^\times M_{\omega} \\ C_{\omega \theta} &= \dot{M}_{\omega \theta} + \omega_0^\times M_{\omega \theta}\end{aligned}\quad (78)$$

The inertia matrix derivatives are

$$\begin{aligned}\dot{M}_{\omega v} &= (\dot{M}_{v\omega})^\top \\ \dot{M}_{\omega} &= \frac{\partial M_{\omega}}{\partial \theta} \dot{\theta} \\ \dot{M}_{\omega \theta} &= \frac{\partial M_{\omega \theta}}{\partial \theta} \dot{\theta}\end{aligned}\quad (79)$$

In Eq. (78), the quantity that results from multiplying the third term in the expression for $C_{\omega v}$ by v_0 will always be zero when the system linear momentum is zero. Since this is the assumption throughout this work, we show this below.

Using Eqs. (60) and (61), the third term in Eq. (78) multiplied by v_0 can be rearranged as

$$\left(\sum_{i=1}^n m_i J_{pi} \dot{\theta} - \sum_{i=1}^n m_i r_0^{ix} \omega_0 \right)^\times R_I^{0\top} v_0 = \left(R_I^{0\top} M_{v\omega} \omega_0 + R_I^{0\top} M_{v\theta} \dot{\theta} \right)^\times R_I^{0\top} v_0 \quad (80)$$

This can be further manipulated as

$$\left(R_I^{0\top} M_{v\omega} \omega_0 + R_I^{0\top} M_{v\theta} \dot{\theta} \right)^\times R_I^{0\top} v_0 = \left(R_I^{0\top} \left(M_{v\omega} \omega_0 + M_{v\theta} \dot{\theta} \right) \right)^\times R_I^{0\top} v_0 \quad (81)$$

$$= R_I^{0\top} \left(M_{v\omega} \omega_0 + M_{v\theta} \dot{\theta} \right)^\times v_0 \quad (82)$$

Now if the linear momentum of the system is zero, then (from Section 3.3.6)

$$M_v v_0 + M_{v\omega} \omega_0 + M_{v\theta} \dot{\theta} = 0 \quad (83)$$

$$M_{v\omega} \omega_0 + M_{v\theta} \dot{\theta} = -M_v v_0 \quad (84)$$

Substituting Eq. (84) into Eq. (82), gives

$$R_I^{0\top} \left(M_{v\omega} \omega_0 + M_{v\theta} \dot{\theta} \right)^\times v_0 = -R_I^{0\top} (M_v v_0)^\times v_0 \quad (85)$$

$$= -m_T R_I^{0\top} v_0^\times v_0 \quad (86)$$

$$= 0 \quad (87)$$

Therefore, throughout the rest of this work, since we will always assume the system linear momentum is zero, we can define $C_{\omega v}$ as just

$$C_{\omega v} = \dot{M}_{\omega v} + \omega_0^\times M_{\omega v} \quad (88)$$

Finally, the EOM associated with $\dot{\theta}$ can be determined using the standard form of Lagrange's equation,

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\theta}} \right) - \frac{\partial T}{\partial \theta} = Q_\theta \quad (89)$$

The first term is

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{\theta}} \right) = \frac{d}{dt} \left(M_{\theta v} v_0 + M_{\theta \omega} \omega_0 + M_{\theta \dot{\theta}} \dot{\theta} \right) \quad (90)$$

$$= M_{\theta v} \dot{v}_0 + M_{\theta \omega} \dot{\omega}_0 + M_{\theta \ddot{\theta}} \ddot{\theta} + \dot{M}_{\theta v} v_0 + \dot{M}_{\theta \omega} \omega_0 + \dot{M}_{\theta \dot{\theta}} \dot{\theta}$$

The second term is

$$\frac{\partial T}{\partial \theta} = v_0^\top \frac{\partial M_{v\omega}}{\partial \theta} \omega_0 + v_0^\top \frac{\partial M_{v\theta}}{\partial \theta} \dot{\theta} + \frac{1}{2} \omega_0^\top \frac{\partial M_{\omega\omega}}{\partial \theta} \omega_0 + \omega_0^\top \frac{\partial M_{\omega\theta}}{\partial \theta} \dot{\theta} + \frac{1}{2} \dot{\theta}^\top \frac{\partial M_{\theta\theta}}{\partial \theta} \dot{\theta} \quad (91)$$

Then the $\dot{\theta}$ row of the C matrix can be defined

$$\begin{aligned} C_{\theta v} &= \dot{M}_{\theta v} \\ C_{\theta \omega} &= \dot{M}_{\theta \omega} - v_0^\top \frac{\partial M_{v\omega}}{\partial \theta} - \frac{1}{2} \omega_0^\top \frac{\partial M_\omega}{\partial \theta} \\ C_\theta &= \dot{M}_\theta - v_0^\top \frac{\partial M_{v\theta}}{\partial \theta} - \omega_0^\top \frac{\partial M_{\omega\theta}}{\partial \theta} - \frac{1}{2} \dot{\theta}^\top \frac{\partial M_\theta}{\partial \theta} \end{aligned} \quad (92)$$

where

$$\begin{aligned} \dot{M}_{\theta v} &= \left(\dot{M}_{v\theta} \right)^\top \\ \dot{M}_{\theta \omega} &= \left(\dot{M}_{\omega\theta} \right)^\top \\ \dot{M}_\theta &= \frac{\partial M_\theta}{\partial \theta} \dot{\theta} \end{aligned} \quad (93)$$

In general, the M and C matrices are fully populated (see Appendix C for the M and C matrices of a two-link planar SMM), so it is clear that there is significant interaction between the arm motion and the base motion. Understanding and controlling this interaction is central to all SMM control strategies.

At this point we have constructed the equations of motion in the form

$$M\ddot{q} + C\dot{q} = Q \quad (94)$$

where q has only been defined in terms of its first time derivative, the generalized velocities \dot{q} . In the case of v_0 and $\dot{\theta}$, two of the three parts that make up \dot{q} , direct integration can provide the base position, r_0 , and the joint positions, θ . However, ω_0 cannot be integrated to find the base attitude. For this, some form of true attitude coordinates, Ω , must be chosen and related to angular velocity. We chose to use Euler parameters (commonly referred to in spacecraft attitude control as quaternions) to avoid the singularity problems associated with Euler angles. Then the equations of

motion given in Eq. (94) can be written as the first order state equations

$$\dot{x}_1 = \begin{bmatrix} U & 0 & 0 \\ 0 & A(\Omega) & 0 \\ 0 & 0 & U \end{bmatrix} x_2 \quad (95)$$

$$\dot{x}_2 = M^{-1}(Q - Cx_2) \quad (96)$$

where x_1 represents the position, orientation and joint position (r_0, Ω, θ) , and x_2 represents the corresponding velocities $(\dot{r}_0, \omega, \dot{\theta})$. The 4×3 matrix $A(\Omega)$ is given by

$$A = \frac{1}{2} \begin{bmatrix} \varepsilon^\times + \eta U \\ \varepsilon^\top \end{bmatrix} \quad (97)$$

where the Euler parameters are $\Omega = (\varepsilon, \eta)$. For more information on Euler parameters, see Hughes [15].

3.6 Momenta of an n -Link SMM

In the analysis of a dynamic system, there are generally a number of fundamental quantities which provide special insight. One such quantity, the system kinetic energy, was used in the previous section to obtain equations of motion using Lagrange's equation. Two other important quantities are the linear momentum and angular momentum. The momenta are particularly useful when no external forces or torques are applied to the system. In this case, the momenta remain constant while the system is in motion. These constants of the motion may in some cases be used to reduce the dimension of the analysis. This section will derive expressions for the momenta in a form similar to the SMM kinematic equations (see Eq. (8)). These expressions, also linear with respect to the generalized velocities, simplify the discussion of singularities and controllers in the subsequent chapters.

3.6.1 Linear Momentum

For a single rigid body, the linear momentum of the body is the product of the body's mass and the inertial linear velocity of the body's center of mass. The total linear momentum of an n -link

SMM is the sum of the linear momentum of each body,

$$\mathbf{p} = \sum_{i=0}^n m_i \mathbf{v}_i \quad (98)$$

Pulling the term corresponding to the base out of the summation, and expanding \mathbf{v}_i in the subsequent terms using Eqs. (48) and (49) to show explicitly the dependence of the link velocity on the base motion, Eq. (98) becomes

$$\mathbf{p} = m_0 \mathbf{v}_0 + \sum_{i=1}^n m_i (\mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^i + \mathbf{v}_0^i) \quad (99)$$

This expression can be put in matrix form by writing \mathbf{v}_0 in \mathcal{F}_I components and $\boldsymbol{\omega}_0$, \mathbf{r}_0^i , and \mathbf{v}_0^i in \mathcal{F}_0 components, following the convention of Sections 3.2 and 3.5. Then the linear momentum is

$$p = m_0 v_0 + \sum_{i=1}^n m_i (v_0 + R_I^0 \omega_0^\times r_0^i + R_I^0 v_0^i) \quad (100)$$

The relative velocity of link i , v_0^i , can be expressed in terms of the joint velocities, $\dot{\theta}$, using the link Jacobians introduced in Eq. (52). Finally, v_0 , ω_0 , and $\dot{\theta}$ can be moved outside the summation to obtain the desired expression for the system linear momentum,

$$p = P_v v_0 + P_\omega \omega + P_\theta \dot{\theta} \quad (101)$$

where

$$\begin{aligned} P_v &= \left(m_0 + \sum_{i=1}^n m_i \right) U_{3 \times 3} = m_T U_{3 \times 3} \\ P_\omega &= -R_I^0 \sum_{i=1}^n \{ m_i r_0^{i \times} \} \\ P_\theta &= R_I^0 \sum_{i=1}^n \{ m_i J_{pi} \} \end{aligned} \quad (102)$$

3.6.2 Angular Momentum

An analogous expression for the angular momentum of the system can be derived in a similar fashion. For a single rigid body, the angular momentum is formed by integrating the cross product of position and inertial velocity for each differential mass element in the body. The total angular

momentum can be expressed as the sum of the angular momentum of each body,

$$\mathbf{h} = \sum_{i=0}^n \{m_i (\mathbf{r}_i \times \mathbf{v}_i) + \mathbf{I}_i \cdot \boldsymbol{\omega}_i\} \quad (103)$$

As before, pull out the base term and expand the remaining terms,

$$\begin{aligned} \mathbf{h} = & m_0 (\mathbf{r}_0 \times \mathbf{v}_0) + \mathbf{I}_0 \cdot \boldsymbol{\omega}_0 \\ & + \sum_{i=1}^n \{m_i ((\mathbf{r}_0 + \mathbf{r}_0^i) \times (\mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^i + \mathbf{v}_0^i)) + \mathbf{I}_i \cdot (\boldsymbol{\omega}_0 + \boldsymbol{\omega}_0^i)\} \end{aligned} \quad (104)$$

At this point, one could choose appropriate coordinate frames for each variable, introduce the joint velocities using the Jacobian relations, and create a matrix expression similar to Eq. (101). However, before proceeding with these steps, Eq. (104) can be significantly simplified, provided the linear momentum of the system is assumed to be zero. This assumption will be made in every case in which the angular momentum is of interest.⁴ To make the simplification, first collect terms containing \mathbf{r}_0

$$\begin{aligned} \mathbf{h} = & \mathbf{r}_0 \times \left(m_0 \mathbf{v}_0 + \sum_{i=1}^n \{m_i (\mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^i + \mathbf{v}_0^i)\} \right) \\ & + \mathbf{I}_0 \cdot \boldsymbol{\omega}_0 + \sum_{i=1}^n \{m_i (\mathbf{r}_0^i \times (\mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^i + \mathbf{v}_0^i)) + \mathbf{I}_i \cdot (\boldsymbol{\omega}_0 + \boldsymbol{\omega}_0^i)\} \end{aligned} \quad (105)$$

Comparing Eq. (105) with Eq. (99), it can be seen that the quantity crossed by \mathbf{r}_0 in the first term is the system linear momentum, \mathbf{p} . The remaining terms are the angular momentum of the system with respect to the satellite base, and can be denoted by \mathbf{h}_0 . Then Eq. (105) becomes

$$\mathbf{h} = \mathbf{r}_0 \times \mathbf{p} + \mathbf{h}_0 \quad (106)$$

Since $\mathbf{p} = 0$, the first term of Eq. (106) is zero, and the system angular momentum is

$$\mathbf{h} = \mathbf{h}_0 = \mathbf{I}_0 \cdot \boldsymbol{\omega}_0 + \sum_{i=1}^n \{m_i (\mathbf{r}_0^i \times (\mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{r}_0^i + \mathbf{v}_0^i)) + \mathbf{I}_i \cdot (\boldsymbol{\omega}_0 + \boldsymbol{\omega}_0^i)\} \quad (107)$$

⁴If there are no external forces or torques, the linear and angular momentum are constant. Neglecting environmental effects, this is the expected condition for free-floating SMMs. Assuming the SMM begins at rest, both momenta will be zero throughout the completion of a given task. However, in real systems environmental forces and torques are sure to exist, so some means of periodically nulling momentum must be available. For SMMs where the base is controlled, the linear and angular momentum may not remain zero, but in this case the expression for angular momentum is not used (see Chapters 4-6).

Now writing this expression in matrix form, again with \mathbf{v}_0 in \mathcal{F}_I components, $\boldsymbol{\omega}_0$, \mathbf{r}_0^i , $\boldsymbol{\omega}_0^i$, and \mathbf{v}_0^i in \mathcal{F}_0 components, and \mathbf{I}_i in \mathcal{F}_i components,

$$h = \left(\sum_{i=1}^n m_i r_0^{i \times} \right) R_I^{0\top} v_0 + \left(I_0 + \sum_{i=1}^n \left\{ R_0^i I_i R_0^{i\top} - m_i r_0^{i \times} r_0^{i \times} \right\} \right) \omega_0 + \sum_{i=1}^n \left\{ R_0^i I_i R_0^{i\top} \omega_0^i + m_i r_0^{i \times} v_0^i \right\} \quad (108)$$

Substituting for ω_0^i and v_0^i using Eq. (52) to put the last summation in terms of θ gives

$$h = H_v v_0 + H_\omega \omega + H_\theta \dot{\theta} \quad (109)$$

where

$$\begin{aligned} H_v &= \left(\sum_{i=1}^n m_i r_0^{i \times} \right) R_I^{0\top} \\ H_\omega &= I_0 + \sum_{i=1}^n \left\{ R_0^i I_i R_0^{i\top} - m_i r_0^{i \times} r_0^{i \times} \right\} \\ H_\theta &= \sum_{i=1}^n \left\{ R_0^i I_i R_0^{i\top} J_{ai} + m_i r_0^{i \times} J_{pi} \right\} \end{aligned} \quad (110)$$

The momenta of the system are closely tied to the equations of motion of the system. The matrices used here to define the momenta are exactly equal to the corresponding submatrices of the inertia matrix used in determining the equations of motion. That is,

$$\begin{aligned} P_v &= M_v \\ P_\omega &= M_{v\omega} \\ P_\theta &= M_{v\theta} \\ H_v &= M_{\omega v} \\ H_\omega &= M_\omega \\ H_\theta &= M_{\omega\theta} \end{aligned} \quad (111)$$

Generally, when writing the system momentum, we use the P and H matrices, and when referring to the equations of motion, we use the M matrices.

3.7 Summary

In Chapter 3, we established the framework for our analysis of SMM control. The kinematic equations for a fixed-base manipulator were developed to show the relation of the end-effector motion to the motion of the arm joints. These equations were extended to the kinematics of an n -link SMM, introducing the effect of base motion. Next, the equations of motion were derived using an energy approach, first for the fixed-base manipulator and then for the n -link SMM. The spacecraft attitude equations were developed using a quasi-coordinate form of Lagrange's equations.⁵ Finally, expressions for the total linear and angular momenta of an n -link SMM were derived. The equations of motion are used as the plant model in the controller and simulation developments of Chapters 5 and 6, and the momentum expressions are key to the development of momentum-constrained Jacobians in Chapter 4.

⁵This was the first derivation to include the body frame components of base angular velocity in the equations of motion. Earlier researchers avoided the use of quasi-coordinates by doing the planar case only, using Euler angle derivatives, or eliminating the angular velocity with the momentum constraint before applying Lagrange's Equation.

Chapter 4 - Singularities of Momentum-Constrained Jacobians

In Chapter 3, the kinematics and dynamics of an n -link SMM were developed. In this chapter, the concept of the manipulator Jacobian is extended to SMMs with varying levels of base control. Some of these new Jacobians have singularities with properties different from the singularities associated with fixed-base manipulator Jacobians. The effects of these singularities are investigated, and methods of alleviating associated problems are demonstrated.

4.1 Free-Floating SMMs: The Generalized Jacobian Matrix and Dynamic Singularities

For free-floating⁶ manipulators, the system dynamics play an integral role in the traditionally kinematic problem of relating end-effector motion and joint motion. Umetani and Yoshida were the first to propose the concept of using momentum conservation equations to eliminate the satellite base motion variables from the kinematic equations to create a new type of manipulator Jacobian. They termed the new Jacobian the Generalized Jacobian Matrix (GJM). Several subsequent controllers have been based upon this concept [23, 28, 51]. Papadopoulos and Dubowsky [35] raised concerns about these methods by noting the existence of dynamic singularities—SMM configurations where the GJM is rank deficient. They showed that these represent configurations at which the end-effector is physically unable to move in the singular direction. They divided the reachable workspace of an SMM into two parts: the path independent workspace (PIW) and the path dependent workspace (PDW). The PIW consists of all points in workspace in which there is no possibility of encountering a dynamic singularity, whereas the PDW encompasses all of the remaining workspace. In the PDW, every end-effector position can be associated with a dynamic singularity,

⁶The exact meaning of the terms “free-floating” and “free-flying” varies between authors in the space robotics literature. A free-floating SMM is defined here to mean an SMM with no active base control in either translation or orientation. The adjective free-flying generally indicates that the robot is mounted on a base capable of moving freely in the workspace. Since this is assumed in the definition of a satellite-mounted manipulator, this term is seldom used in this work.

although whether a singularity actually occurs during any given task depends upon the path of the manipulator motion. They concluded that nearly any control algorithm derived for terrestrial robots could be used for SMMs if the dynamic singularities are avoided.

To construct the Generalized Jacobian Matrix, we begin with the expression for the velocity kinematics which was derived in Chapter 3 (Eq. (11)),

$$\dot{r} = J_v v_0 + J_\omega \omega + J_\theta \dot{\theta} \quad (112)$$

For a fixed-base manipulator, the base motion is zero, and the first two terms of the equation disappear. At any given instant, the end-effector velocity depends only on the joint velocities, which may be directly controlled through the joint actuators. This direct link between the system input and output is highly desirable when constructing the robot controller. For a free-floating SMM, the base motion terms of Eq. (112) are not zero, but no base actuation exists. The lack of direct actuation of the kinematic variables in Eq. (112) provides the motivation for constructing the GJM.

A Jacobian that maps joint velocity space to end-effector velocity space can be constructed using conservation of momentum to eliminate v_0 and ω from Eq. (112). The momentum expressions were derived in Chapter 3 as Eqs. (101) and (109) and are repeated here, where we assume the momenta are zero for a free-floating SMM

$$p = P_v v_0 + P_\omega \omega + P_\theta \dot{\theta} = 0 \quad (113)$$

$$h = H_v v_0 + H_\omega \omega + H_\theta \dot{\theta} = 0 \quad (114)$$

Equation (113) can be solved for v_0 ,

$$v_0 = -\frac{1}{m_T} (P_\omega \omega + P_\theta \dot{\theta}) \quad (115)$$

Substituting this expression for v_0 into Eq. (114) gives

$$h = \left(H_\omega - \frac{1}{m_T} H_v P_\omega \right) \omega + \left(H_\theta - \frac{1}{m_T} H_v P_\theta \right) \dot{\theta} \quad (116)$$

$$= \tilde{H}_\omega \omega + \tilde{H}_\theta \dot{\theta} \quad (117)$$

Using the expressions for the H and P matrices given in Eqs. (102) and (110), \tilde{H}_ω and \tilde{H}_θ are

$$\tilde{H}_\omega = I_0 + \sum_{i=1}^n \{R_0^{iT} I_i R_0^i - m_i r_0^{i \times} r_0^{i \times}\} - \frac{1}{m_T} \sum_{i=1}^n m_i r_0^{i \times} \sum_{i=1}^n m_i r_0^{i \times} \quad (118)$$

$$\tilde{H}_\theta = \sum_{i=1}^n \{R_0^{iT} I_i R_0^i J_{\omega i} + m_i r_0^{i \times} J_{v i}\} - \frac{1}{m_T} \sum_{i=1}^n m_i r_0^{i \times} \sum_{i=1}^n m_i J_{v i} \quad (119)$$

Now recalling that $h = 0$, Eq. (117) can be solved for ω ,

$$\omega = -\tilde{H}_\omega^{-1} \tilde{H}_\theta \dot{\theta} \quad (120)$$

Combining with Eqs. (112), (113), (115), and (120) gives

$$\dot{r}_e = \left(\frac{1}{m_T} J_v (P_\omega \tilde{H}_\omega^{-1} \tilde{H}_\theta - P_\theta) - J_\omega \tilde{H}_\omega^{-1} \tilde{H}_\theta + J_\theta \right) \dot{\theta} \quad (121)$$

$$= \tilde{J} \dot{\theta} \quad (122)$$

The Jacobian \tilde{J} is equivalent to the Generalized Jacobian Matrix first developed by Umetani and Yoshida. Satellite-mounted manipulator configurations where \tilde{J} is less than full row rank, were termed dynamic singularities by Papadopoulos and Dubowsky, since \tilde{J} depends not only on the system configuration, but on the inertial properties as well. The dimensions of \tilde{J} will depend upon the sizes of \dot{r}_e and $\dot{\theta}$. If \tilde{J} is square, dynamic singularities are equivalent to

$$\det(\tilde{J}) = 0 \quad (123)$$

Since \tilde{J} is frequently not square in subsequent sections, we use the condition

$$\underline{\sigma}(\tilde{J}) = 0 \quad (124)$$

where $\underline{\sigma}(\cdot)$ denotes the minimum singular value, to find singular configurations by a numerical search.

Papadopoulos and Dubowsky also noted that dynamic singularities do not depend on the spacecraft position or attitude. This can be seen by examining each of the terms in Eq. (121). Definitions of these variables are given in Eqs. (9), (102), (118), and (119). There is no dependence on spacecraft position, r_0 , in the definition of the GJM, and \tilde{J} depends on the spacecraft base attitude only through the rotation matrix R_1^0 . This rotation matrix can be factored out in front of all terms that

make up \tilde{J} , giving

$$\tilde{J} = R_I^0 \hat{J}(\theta) \quad (125)$$

Since the rotation matrix is nonsingular, all singularities of \tilde{J} must be contained in \hat{J} , which depends only on the joint configuration.

Recall that the end-effector position in workspace depends on not only the joint configuration, but on the spacecraft position and orientation as well. Consequently, a distinct singular joint configuration, represented by a (zero-dimensional) point θ_s in joint space, corresponds to a higher dimensional region in inertial workspace. Using a spatial example, imagine holding the arm joints fixed at a singular configuration ($\theta = \theta_s$) and varying r_0 and Ω in Eq. (5) over the complete range of possible values. The end-effector is mapped to a sphere in workspace, centered about the center of mass of the system. This sphere is a two-dimensional representation in workspace of all the possible end-effector positions that can be associated with that particular singular configuration, represented by a zero-dimensional point in joint space.

The dynamic singularities of an SMM bear a close relationship to the kinematic singularities of its arm. The kinematic singularities form a set of manifolds in joint space, and the dynamic singularities form similar sets. Each manifold of dynamic singularities is a perturbation of a corresponding manifold of kinematic singularities. As the ratio of manipulator mass/inertia to base mass/inertia increases, the size of this perturbation increases. The reverse is also true. Indeed, examining the definition of \tilde{J} in Eq. (121), it is clear that as \tilde{H}_ω and m_T grow relative to \tilde{H}_θ and P_θ , \tilde{J} approaches J_θ . This correspondence between kinematic and dynamic singularities motivates much of the following discussion, in which we seek to solve problems caused by dynamic singularities using methods similar to those which we would use to handle problems stemming from kinematic singularities.

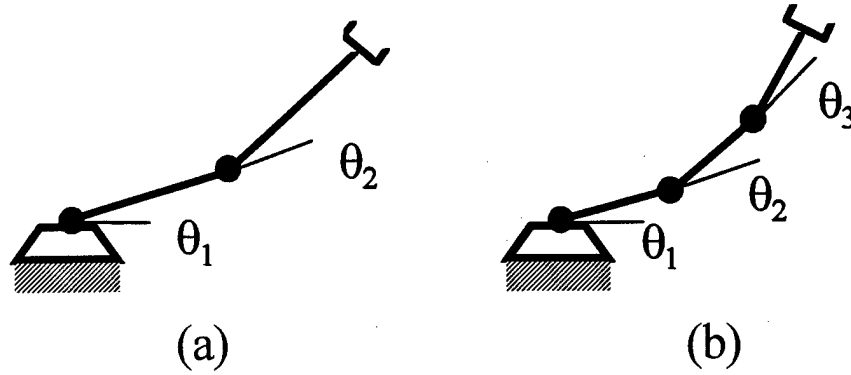


Figure 4. Fixed-Base Planar Arms, Two-Link (a) and Three-Link (b)

These characteristics of the GJM and its singularities have all been expressed or implied in the work of Umetani and Yoshida, Papadopoulos and Dubowsky, and others. In the remainder of this chapter, we investigate some means of reducing the effects of dynamic singularities.

4.2 The Impact Of Redundancy

For terrestrial robots, kinematic singularities are an important consideration in path planning and control. Many control methods rely upon inverting the manipulator Jacobian (such as the resolved rate [50] and resolved acceleration control methods [22]). In the neighborhood of a singularity, these algorithms generate high command rates/accelerations for the joints and can fail completely at singularities. Redundancy has an interesting impact on this problem. For many designs, increased degrees of freedom results in an increase in the number of singular configurations. As an example, consider the difference between the planar two-link arm and the redundant planar three-link arm, shown in Figure 4. The two-link arm is singular when $\theta_2 = \pm k\pi$ ($k = 0, 1, 2, \dots$), corresponding to the outer and inner boundaries of its workspace. However, the three-link arm is singular when $\theta_2 = \pm k\pi$ and $\theta_3 = \pm k\pi$ ($k = 0, 1, 2, \dots$), corresponding not only to workspace boundaries, but also to interior points. The increased number of singularities is offset by the ability

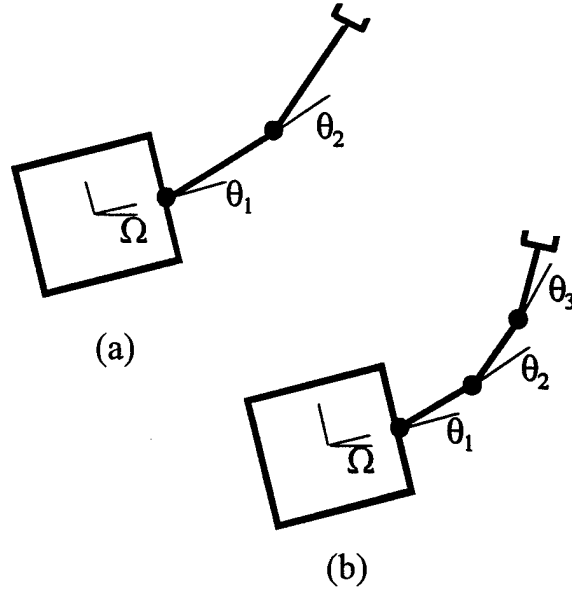


Figure 5. Planar SMMs, Two-Link (a) and Three-Link (b)

of the redundant arm to “maneuver around” interior singularities using null motion.⁷ Interior points in workspace can be reached by many joint configurations, only a few of which may be singular.

For space manipulators, the effect of redundancy is quite similar, increasing potential singularities while providing a means for singularity avoidance. The singularity increase is apparent when one considers the correspondence of singularity manifolds discussed earlier. Since the redundancy in the arm creates new kinematic singularity manifolds, the SMM has new dynamic singularity manifolds as well, each a perturbation of its kinematic counterpart. The means of singularity avoidance is identical—the SMM with a redundant arm has null motion associated with the extra degree of freedom (DOF).

To illustrate, compare the two planar SMMs shown in Figure 5. The first is a spacecraft base with a two-link arm, and the second is a base with a three-link arm. The physical parameters of the two-link are given in Table 1, and those of the three-link are given in Table 2. All joints are revolute.

⁷Null motion is defined as joint motion which does not alter the position of the end-effector in workspace. It is a product of the redundancy of the system and is directly associated with the null space of the manipulator Jacobian.

Body	l_i (m)	m_i (kg)	I_i (kg · m ²)
0	1.0	100	16.67
1	1.0	20	1.733
2	1.0	20	1.733

Table 1. Two-Link SMM Physical Parameters

Both bases are identical, as is the total mass and length of the manipulators, so that the reachable workspace is equal as well. The only difference is the additional DOF in the three-link arm. Using the condition given in Eq. (124), the singular joint configurations can be found through a numerical search of joint space. The corresponding end-effector positions in inertial workspace can be found with Eq. (5), letting r_0 and Ω vary over all possible combinations.

The inertial workspace of the two-link is shown in Figure 6, with the regions of potential singularities (Papadopoulos' PDW) shaded. The two bands of the PDW can be associated with the two kinematic singularities of a fixed-base two-link arm. The outer band consists of the set of dynamic singularities which include the kinematic singularities $\theta_2 = 2k\pi$ ($k = 0, 1, 2, \dots$), whereas the inner band consists of a second set of dynamic singularities which include the kinematic singularities $\theta_2 = (2k + 1)\pi$ ($k = 0, 1, 2, \dots$).

The inertial workspace of the three-link is shown in Figure 7. In this case, a much greater portion of the total workspace is part of the PDW. This is a result of the added sets of dynamic singularities. Each new singularity set can be associated with one of the internal kinematic singularities of the fixed-base three-link. The expanded size of the PDW suggests that the redundant SMM is more likely to suffer from the negative effects of dynamic singularities. However, singularity avoidance methods used for fixed-base redundant robots can be extended to space manipulators.

The typical singularity avoidance method is based on using the system redundancy. Consider the task of controlling the end-effector position in a plane. This requires the velocity kinematic relation given in Eq. (122). For the nonredundant two-link arm, the Generalized Jacobian, \tilde{J} , is

Body	l_i (m)	m_i (kg)	I_i ($\text{kg} \cdot \text{m}^2$)
0	1.0	100	16.67
1	0.67	13.33	0.5383
2	0.67	13.33	0.5383
3	0.67	13.33	0.5383

Table 2. Three-Link SMM Physical Parameters

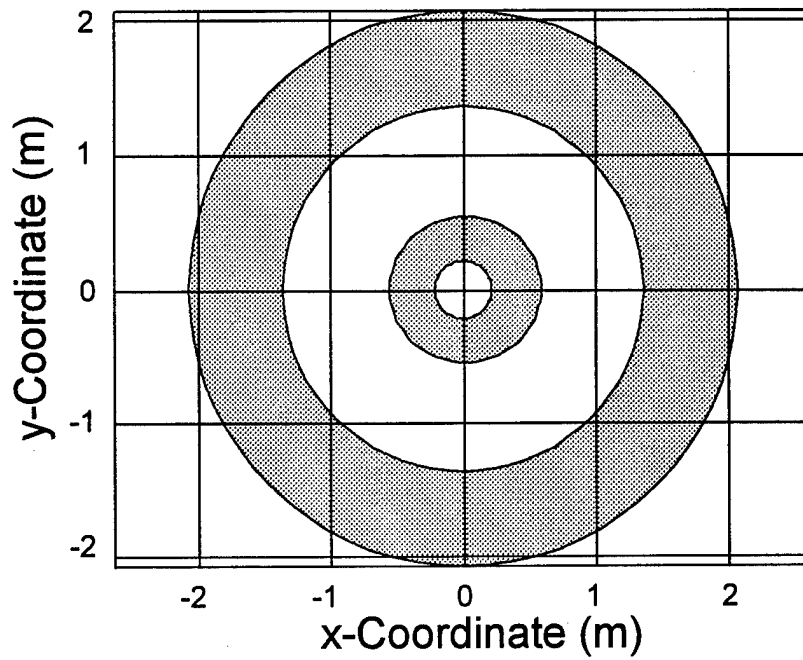


Figure 6. Inertial Workspace of Two-Link SMM

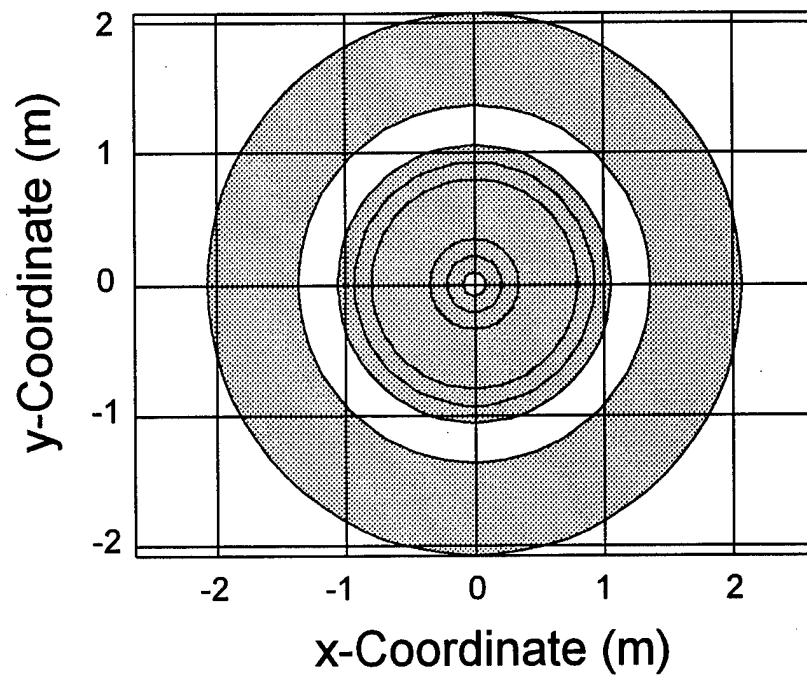


Figure 7. Inertial Workspace of Three-Link SMM

2×2 , and can be inverted to determine joint velocity commands that give the desired end-effector velocity,

$$\dot{\theta}_c = \tilde{J}^{-1} \dot{r}_c \quad (126)$$

When the manipulator is in the neighborhood of a dynamic singularity, this relation results in large joint velocity commands, and will fail completely at a dynamic singularity. However, for the redundant three-link arm, \tilde{J} is 2×3 and the inversion of Eq. (122) requires a pseudoinverse,⁸

$$\dot{\theta}_c = \tilde{J}^\# \dot{r}_c \quad (127)$$

The pseudoinverse has the advantage that it is defined even if \tilde{J} is singular, but it still results in large rates near the dynamic singularities, since the pseudoinverse returns an exact solution when it exists. However, the pseudoinverse does not give a unique solution to Eq. (122). Since \tilde{J} is 2×3 and has rank two when not at a dynamic singularity, it has a null space of dimension one. The solutions to Eq. (122) can be parameterized by

$$\dot{\theta}_c = \tilde{J}^\# \dot{r}_c + (U - \tilde{J}^\# \tilde{J})z \quad (128)$$

where $(U - \tilde{J}^\# \tilde{J})$ is referred to as the null space projection operator. Varying the parameter z gives all possible choices of $\dot{\theta}_c$ which result in the desired end-effector velocity. Liegeois [17] first demonstrated how this type of relation can be used to reduce a potential function by substituting the gradient of the potential for the parameter z ,

$$\dot{\theta}_c = J^\# v_{ed} + (U - J^\# J) \left(-k \frac{\partial F}{\partial \theta} \right)^T \quad (129)$$

where $F = F(\theta)$ is a potential function, k is a positive gain factor, and J is the manipulator Jacobian. Yoshikawa [52] applied this method to singularity avoidance of terrestrial robots by defining a potential function based on his measure of manipulability,

$$F = -\sqrt{\det(JJ^T)} \quad (130)$$

⁸Recall that by "pseudoinverse" we mean the Moore-Penrose Generalized Inverse, computed using a singular value decomposition (SVD) based method.

This potential function has a maximum at kinematic singularities, so using Eq. (129) to reduce the potential function steers the system away from singularities and towards configuration where manipulability is high.

We translated these ideas to space manipulators by replacing the manipulator Jacobian, J , with the Generalized Jacobian, \tilde{J} . Although it is possible to use a potential like the one in Eq. (130), the complexity of the Generalized Jacobian makes it impractical to compute the gradient of this type of potential function analytically. An alternative is to define a simple potential function with similar maxima. A simple choice for the three-link SMM is

$$F(\theta) = \frac{1}{(\epsilon + \sin^2 \theta_2)(\epsilon + \sin^2 \theta_3)} \quad (131)$$

This function was chosen because, for base-to-arm mass ratios on the order of our example, the dynamic singularities of the three-link SMM all occur when θ_2 and θ_3 are near zero or π . The constant ϵ is a small positive number used to keep the potential function finite for all values of θ . The analytic expression for the gradient is easily found and has the added benefit of being simple to compute each time it is needed by the control algorithm.

4.3 Prismatic Joints

The previous section demonstrated that designing redundancy into an SMM provides mixed results in terms of eliminating dynamic singularities. Another approach is to use prismatic joints in place of revolute joints. Motivation for this concept stems from the usefulness of prismatic joints in eliminating kinematic singularities in fixed-base manipulators. Consider two designs for a two DOF planar manipulator, the two-link arm from the previous section (Figure 4), and the one-link arm shown in Figure 8. The single link manipulator has collocated revolute and prismatic joints which provide a total workspace identical to the two-link arm. In the remainder of this paper, these manipulator designs will be designated as the revolute-revolute (RR) and revolute-prismatic (RP)

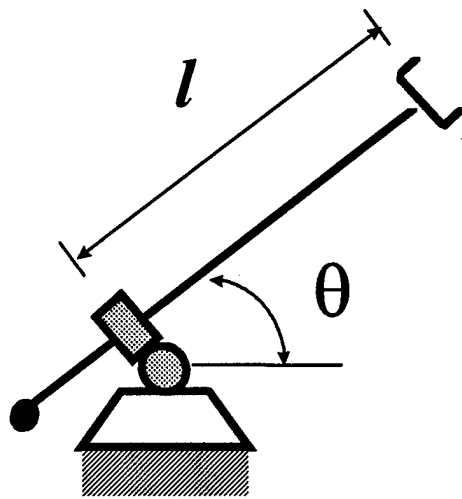


Figure 8. Planar Two-DOF Arm (RP)

designs, respectively. As noted earlier, the RR arm is singular when $\theta_2 = k\pi$, ($k = 0, 1, 2, \dots$). The corresponding end-effector positions include the circle which defines the outer edge of workspace and the central point in the workspace. Now examine the manipulator Jacobian of the RP arm,

$$J_{RP} = \begin{bmatrix} \cos \theta & -l \sin \theta \\ \sin \theta & l \cos \theta \end{bmatrix} \quad (132)$$

It is evident that the Jacobian only loses rank at one point in joint space, $l = 0$. This corresponds to the center of the workspace. Whereas the outer edge of workspace is a kinematic singularity for the RR arm, the Jacobian of the RP arm is still full rank at the outer edge ($l = l_{\max}$). Of course, no motion in a positive radial direction is possible because of the joint limit on the prismatic joint, but motion in a negative radial direction is perfectly feasible. More important is the behavior of the arms *near* the edge. For the RR arm to generate radial motion of the end-effector, large joint velocities are required, whereas the RP arm need not use excessive joint velocities to generate radial motion. In essence, the RP design has eliminated one of the kinematic singularities.

An SMM using a prismatic joint enjoys similar benefits. Consider the one-link, two-DOF SMM shown in Figure 9. Its physical parameters are shown in Table 3. By design, the total reachable workspace is identical to the RR and RRR SMMs shown in Figure 5. The inertial workspace of the RP SMM is shown in Figure 10, where again the shaded region is the portion of workspace in which dynamic singularities may occur (the PDW). Compare this to the workspace of the two-link and three-link SMM in Figures 6 and 7. Clearly, the PDW of the RP design is far smaller than for either the RR or RRR designs. This is primarily a result of eliminating the entire set of singularities at the outer edge of workspace. The remaining region of potential dynamic singularities is created by rotating the dynamic singularity manifold in joint space about the system center of mass. This manifold is a perturbation of the kinematic singularity mentioned for the fixed-base RP manipulator at $l = 0$.

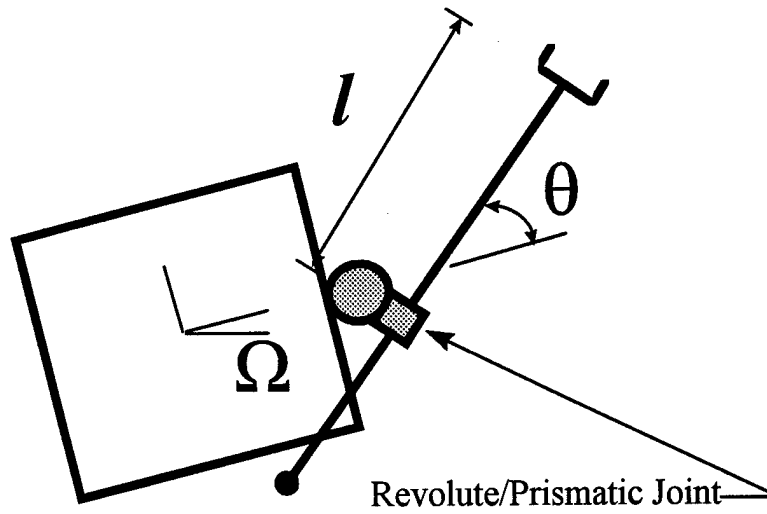


Figure 9. Planar SMM with Two DOF (RP) Arm

Body	l (m)	m_i (kg)	I_i ($kg \cdot m^2$)
0	1.0	100	16.67
1	2.0	40	13.467

Table 3. One-Link (RP) SMM Physical Parameters

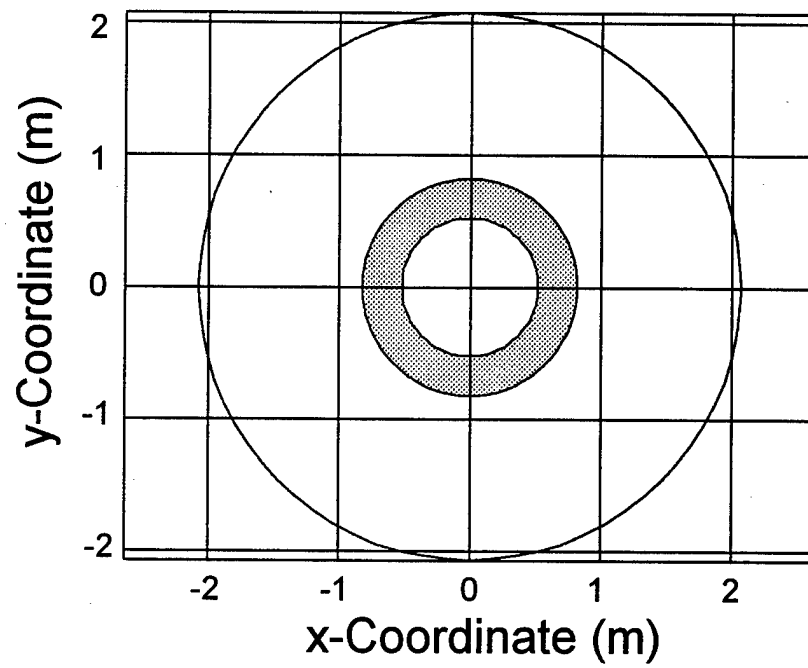


Figure 10. Inertial Workspace of Planar SMM with RP Arm

4.4 Joint Limits

In real mechanisms, joints nearly always have limited range of motion. Revolute joints will rarely rotate a full 2π radians, and prismatic joints must have finite ranges. Joint limits are generally a handicap of real mechanisms, since they typically shrink workspace and decrease manipulability, but a positive effect is that limits can eliminate potentially singular configurations. Consider the RR planar manipulator. If the joint motion is unlimited, then the manipulator has kinematic singularities at $\theta_2 = \pm k\pi$, but if the second joint is restricted to the range, $-\pi/8 < \theta_2 < 7\pi/8$, then only one singularity remains, $\theta_2 = 0$. For the RP manipulator, limiting the prismatic joint such that $l > 0$ eliminates the only kinematic singularity. This might seem unremarkable, since singularities are eliminated by restricting the reachable workspace, essentially just “cutting away” the region that causes the difficulties. But examine the effect of a similar joint limit for the two-link (RR) SMM. Suppose the two-link (RR) SMM has the joint limit $-\pi/2 < \theta_1 < \pi/2$. The reachable workspace is unchanged, since the system is free to rotate about its center of mass. The size of the PDW, however, is reduced as a result of eliminating potential singular configurations. This reduction is shown in Figure 11. Since joint limits can introduce problems as well, one cannot conclude from this example that joint limits are a panacea for dynamic singularities. The problem of avoiding dynamic singularities is traded for the problem of avoiding joint limits. This trade may be beneficial given the proper controller, however, so the role of joint limits in eliminating dynamic singularities should not be overlooked in SMM design.

4.5 A Singularity-Free Design

Previous sections concentrated on single design features and their effects on dynamic singularities. Alone, each feature offered limited advantages, but in combination they can completely eliminate dynamic singularities. Consider the simple example problem used throughout the previous

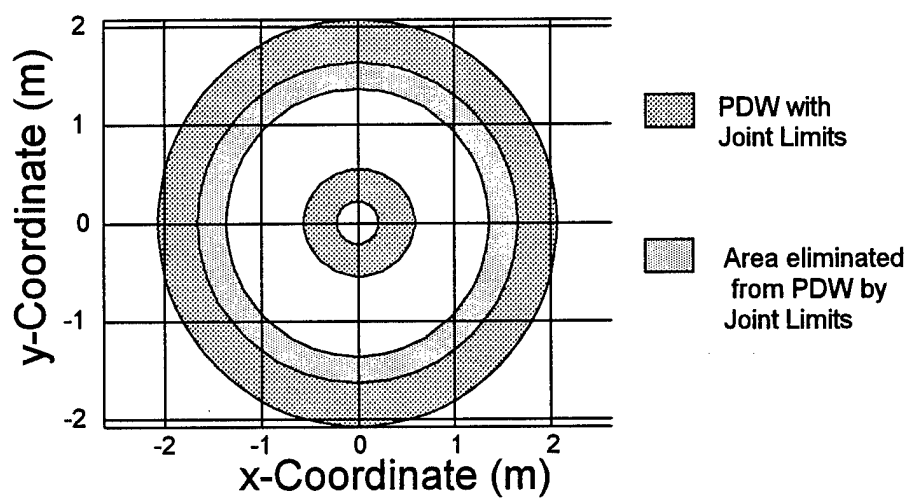


Figure 11. Reducing PDW Using Joint Limits

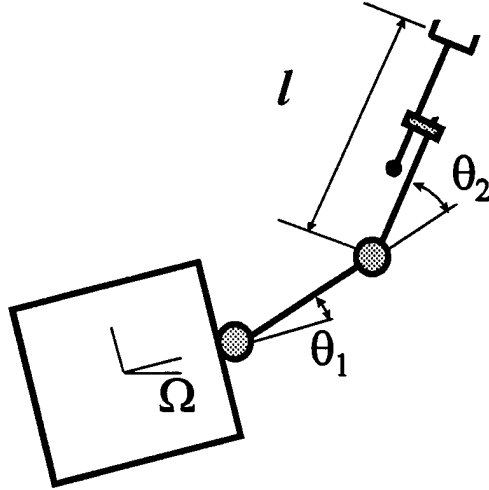


Figure 12. Planar SMM with RRP Arm

sections, that of positioning the end-effector in a plane. An SMM design incorporating redundancy, a prismatic joint, and joint limits is shown in Figure 12. This RRP design resembles the original RR design, but by adding the prismatic joint, the dynamic singularities associated with the outstretched arm ($\theta_2 \approx 0$) and the folded arm ($\theta_2 \approx \pi$) are eliminated. The RRP design also resembles the RP design, but limiting the minimum forearm length by requiring that $l > l_{\min}$ eliminates the singularities near $l = 0$. In the original RP design, this limit would eliminate reachable workspace, but the extra revolute joint allows the limit on the prismatic joint without restricting workspace. Using physical parameters identical to the RRR three-link (Table 2), the reachable workspace is the same as for all of the earlier examples. However, the entire workspace is free of dynamic singularities. In addition, the redundant DOF can be used for avoiding the joint limits. This could be implemented using Eq. (129) with a potential function based on the prismatic joint position

$$F(l) = \frac{1}{(l - l_{\min} + \epsilon)(l - l_{\max} - \epsilon)} \quad (133)$$

This potential function would tend to keep the prismatic joint in the center of its range of motion whenever possible.

It should be noted that although this design has no dynamic singularities and suffers no adverse effects from being near a joint limit, it can still be trapped if the prismatic joint actually hits its limits ($0.667 < l < 1.333$). For this reason, it can be viewed as an improvement over the free-floating revolute designs, but not as the ideal SMM control concept.

4.6 The Impact of Base Control

4.6.1 Full Base Control

Perhaps the most effective means of eliminating the problems caused by dynamic singularities is to control the satellite base of the SMM system. Recall that dynamic singularities are a direct result of a freely moving base. At a singularity, the system is in a configuration in which some direction of end-effector motion is physically unrealizable. In these cases, the reactive motion of the base induces end-effector velocities which exactly cancel the velocity produced by the arm joint motion. Given this, it is unsurprising that controlling the base motion eliminates dynamic singularities. This effect can be seen by examining the relevant equations. Consider an SMM with an n -DOF arm, where the end-effector position and orientation are to be controlled in three dimensions. The velocity kinematics equation, Eq. (112), is

$$\dot{r}_e = J_v v + J_\omega \omega + J_\theta \dot{\theta} \quad (134)$$

Here \dot{r}_e is 6×1 , v and ω are 3×1 , $\dot{\theta}$ is $n \times 1$, J_v and J_ω are 6×3 , and J_θ is $6 \times n$. If the base is controlled in translation and rotation, then momentum is not conserved, and the singularities depend only on the kinematic equation. The system is singular if

$$\text{rank} \left(\begin{bmatrix} J_v & J_\omega & J_\theta \end{bmatrix} \right) < 6 \quad (135)$$

The first six columns of this matrix are

$$\begin{bmatrix} J_v & J_\omega \end{bmatrix} = \begin{bmatrix} U_{3 \times 3} & -r_0^{ex} \\ 0_{3 \times 3} & U_{3 \times 3} \end{bmatrix} \quad (136)$$

which is clearly always of rank 6, so the system cannot be singular. The end-effector can be given an arbitrary velocity by holding the arm joints fixed, and controlling the base. With the joints fixed, the SMM becomes a single rigid body—all points in the body have the same angular velocity, and the translation at the base can be chosen to compensate for translation due to the rotation and produce an arbitrary end-effector translation.

We illustrate this argument for a planar case, in which the end-effector position and angle are controlled, and the SMM consists of a base and three-DOF planar manipulator. In this case, \dot{r}_e and $\dot{\theta}$ are 3×1 , v is 2×1 , and ω is 1×1 . The Jacobians J_v , J_ω , and J_θ are 3×2 , 3×1 , and 3×1 , respectively. The system is singular if

$$\text{rank} \left(\begin{bmatrix} J_v & J_\omega & J_\theta \end{bmatrix} \right) < 3 \quad (137)$$

The first three columns are

$$\begin{bmatrix} J_v & J_\omega \end{bmatrix} = \begin{bmatrix} 1 & 0 & -r_{0y}^e \\ 0 & 1 & r_{0x}^e \\ 0 & 0 & 1 \end{bmatrix} \quad (138)$$

so the system is never singular. The system is completely controllable by base actuation alone.

Consider the example shown in Figure 13, where the SMM joint configuration is $\theta_1 = -\pi/4$, $\theta_2 = \theta_3 = \pi/4$ (refer to Figure 5 for joint angle convention), and the end-effector is tasked to move in a positive x-direction while rotating counter-clockwise. The joints can remain completely fixed and the task can be performed by rotating and translating the base as shown.

While base control is an effective means of eliminating dynamic singularities, it does have an important disadvantage. Controlling the base requires some form of base actuators. For rotation, there are a number of reasonable choices, including thrusters, reaction wheels, and control moment gyros. For translation, thrusters are the only practical alternative. Unfortunately, thrusters require fuel, and conserving fuel is a high priority for all spacecraft. The high cost of base actuation in strategies relying on base control is the most often used argument in support of free-floating strategies.

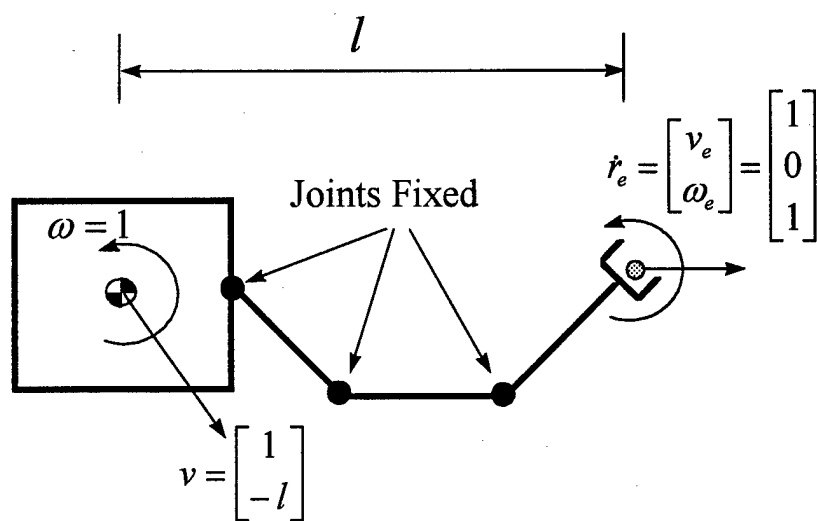


Figure 13. Equivalent Base Motion and End-Effector Motions With Joints Fixed

4.6.2 Base Attitude Control

Another possibility is that of controlling only the base attitude. This eliminates the need for thrusters, since base translation actuation is unnecessary, and base attitude actuation can be done with reaction wheels or control moment gyros. The energy source for these devices is renewable, so the argument for free-floating control is much weaker, especially in light of the resulting performance enhancements that will be shown in this section.

If the RW or CMG cluster is considered "outside" the SMM system, then angular momentum of the system is not conserved. Since no external forces are applied, linear momentum is conserved, and the linear momentum equation still can be used to eliminate kinematic variables. The relevant equations, Eqs. (112) and (113), are

$$\dot{r} = J_v v + J_\omega \omega + J_\theta \dot{\theta} \quad (139)$$

$$p = P_v v + P_\omega \omega + P_\theta \dot{\theta} \quad (140)$$

We eliminate the base translation velocity, v , by solving the linear momentum equation for v (assuming $p = 0$), and substituting into the kinematic velocity equation. This gives

$$\dot{r} = (J_\omega - J_v P_v^{-1} P_\omega) \omega + (J_\theta - J_v P_v^{-1} P_\theta) \dot{\theta} \quad (141)$$

$$= \begin{bmatrix} \tilde{J}_\omega & \tilde{J}_\theta \end{bmatrix} \begin{bmatrix} \omega \\ \dot{\theta} \end{bmatrix} \quad (142)$$

Although the structure of this Jacobian resembles that of the Generalized Jacobian Matrix (see Eq. (121)), it is similar to the manipulator Jacobian of a robot formed by mounting the SMM arm on a spherical joint. Its singularities are essentially kinematic singularities. This can be shown by an example. For the two-link planar SMM, we derive the Jacobian of Eq. (142).

The end-effector position is given by the equation

$$r = r_0 + b_0 + a_1 + b_1 + a_2 + b_2 \quad (143)$$

where r_0 is the position vector of the base center of mass relative to the inertial origin, a_i is the position vector of the i^{th} link center of mass relative to the i^{th} joint, and b_i is the position vector of

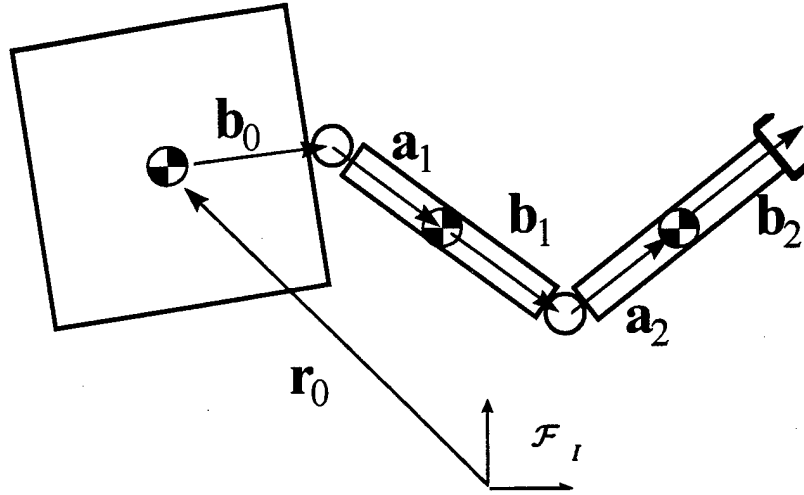


Figure 14. Two-Link Planar SMM

the $i + 1^{th}$ joint relative to the i^{th} center of mass. These vectors are shown in Figure 14. Equation (143) can be written in component form (inertial frame) as

$$r_x = r_{0x} + b_0 c_1 + (a_1 + b_1) c_{12} + (a_2 + b_2) c_{123} \quad (144)$$

$$r_y = r_{0y} + b_0 s_1 + (a_1 + b_1) s_{12} + (a_2 + b_2) s_{123} \quad (145)$$

where we use the following shorthand notation for sine and cosine functions,

$$c_1 \triangleq \cos(\theta_1) \quad (146)$$

$$s_1 \triangleq \sin(\theta_1) \quad (147)$$

$$c_{12} \triangleq \cos(\theta_1 + \theta_2) \quad (148)$$

$$c_{123} \triangleq \cos(\theta_1 + \theta_2 + \theta_3) \quad (149)$$

Differentiating with respect to time, this becomes

$$\dot{r}_x = \dot{r}_{0x} - b_0 \dot{\theta}_1 s_1 - (a_1 + b_1) (\dot{\theta}_1 + \dot{\theta}_2) s_{12} - (a_2 + b_2) (\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3) s_{123} \quad (150)$$

$$\dot{r}_y = \dot{r}_{0y} + b_0 \dot{\theta}_1 c_1 + (a_1 + b_1) (\dot{\theta}_1 + \dot{\theta}_2) c_{12} + (a_2 + b_2) (\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3) c_{123} \quad (151)$$

In matrix form this is the velocity kinematics equation,

$$\dot{r} = J_v v + J_\omega \omega + J_\theta \dot{\theta} \quad (152)$$

where

$$v = \begin{bmatrix} \dot{r}_{0x} \\ \dot{r}_{0y} \end{bmatrix} \quad (153)$$

$$\omega = \dot{\theta}_1 \quad (154)$$

$$\dot{\theta} = \begin{bmatrix} \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \quad (155)$$

$$J_v = U_{2 \times 2} \quad (156)$$

$$J_\omega = \begin{bmatrix} -b_0 s_1 - (a_1 + b_1) s_{12} - (a_2 + b_2) s_{123} \\ b_0 c_1 + (a_1 + b_1) c_{12} + (a_2 + b_2) c_{123} \end{bmatrix} \quad (157)$$

$$J_\theta = \begin{bmatrix} -(a_1 + b_1) s_{12} - (a_2 + b_2) s_{123} & -(a_2 + b_2) s_{123} \\ (a_1 + b_1) c_{12} + (a_2 + b_2) c_{123} & (a_2 + b_2) c_{123} \end{bmatrix} \quad (158)$$

The linear momentum of the system is the sum of the linear momentum of each of the three rigid bodies,

$$p = m_0 v_0 + m_1 v_1 + m_2 v_2 \quad (159)$$

The components of p in the inertial frame are

$$p_x = m_0 \dot{r}_{0x} + m_1 \left(\dot{r}_{0x} - b_0 \dot{\theta}_1 s_1 - a_1 (\dot{\theta}_1 + \dot{\theta}_2) s_{12} \right) \quad (160)$$

$$+ m_2 \left(\dot{r}_{0x} - b_0 \dot{\theta}_1 s_1 - (a_1 + b_1) (\dot{\theta}_1 + \dot{\theta}_2) s_{12} - a_2 (\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3) s_{123} \right)$$

$$p_y = m_0 \dot{r}_{0y} + m_1 \left(\dot{r}_{0y} + b_0 \dot{\theta}_1 c_1 + a_1 (\dot{\theta}_1 + \dot{\theta}_2) c_{12} \right) \quad (161)$$

$$+ m_2 \left(\dot{r}_{0y} + b_0 \dot{\theta}_1 c_1 + (a_1 + b_1) (\dot{\theta}_1 + \dot{\theta}_2) c_{12} + a_2 (\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3) c_{123} \right)$$

Converting to the matrix form of the linear momentum equation, this becomes

$$p = P_v v + P_\omega \omega + P_\theta \theta \quad (162)$$

where v , ω , and θ are as above and

$$P_v = m_T U_{2 \times 2} \quad (163)$$

$$P_\omega = \begin{bmatrix} -(m_1 + m_2) b_0 s_1 - (m_1 a_1 + m_2 (a_1 + b_1)) s_{12} - m_2 a_2 s_{123} \\ (m_1 + m_2) b_0 c_1 + (m_1 a_1 + m_2 (a_1 + b_1)) c_{12} + m_2 a_2 c_{123} \end{bmatrix} \quad (164)$$

$$P_\theta = \begin{bmatrix} -(m_1 a_1 + m_2 (a_1 + b_1)) s_{12} - m_2 a_2 s_{123} & -m_2 a_2 s_{123} \\ (m_1 a_1 + m_2 (a_1 + b_1)) c_{12} + m_2 a_2 c_{123} & m_2 a_2 c_{123} \end{bmatrix} \quad (165)$$

Solving Eq. (162) for the base translation and substituting into Eq. (152),

$$\dot{r} = \left(J_\omega - \frac{1}{m_T} P_\omega \right) \omega + \left(J_\theta - \frac{1}{m_T} P_\theta \right) \dot{\theta} \quad (166)$$

$$= \begin{bmatrix} \tilde{J}_\omega & \tilde{J}_\theta \end{bmatrix} \begin{bmatrix} \omega \\ \dot{\theta} \end{bmatrix} = J_{bac} \begin{bmatrix} \omega \\ \dot{\theta} \end{bmatrix} \quad (167)$$

The elements of the base attitude controlled Jacobian can be determined by substituting Eqs. (157), (158), (164), and (165) into Eq. (166),

$$J_{bac} = \begin{bmatrix} -k_1 s_1 - k_2 s_{12} - k_3 s_{123} & -k_2 s_{12} - k_3 s_{123} & -k_3 s_{123} \\ k_1 c_1 + k_2 c_{12} + k_3 c_{123} & k_2 c_{12} + k_3 c_{123} & k_3 c_{123} \end{bmatrix} \quad (168)$$

where

$$\begin{aligned} k_1 &= m_0 b_0 / m_T \\ k_2 &= (m_0 a_1 + (m_0 + m_1) b_1) / m_T \\ k_3 &= (m_T (a_2 + b_2) - m_2 a_2) / m_T \end{aligned} \quad (169)$$

If the SMM base was pinned to an inertially fixed frame, allowing rotation about the base center of mass, but not translation, it would be kinematically equivalent to a fixed-base three-link planar robot. The manipulator Jacobian, J_{3lk} , for this type of robot is

$$J_{3lk} = \begin{bmatrix} -k_4 s_1 - k_5 s_{12} - k_6 s_{123} & -k_5 s_{12} - k_6 s_{123} & -k_6 s_{123} \\ k_4 c_1 + k_5 c_{12} + k_6 c_{123} & k_5 c_{12} + k_6 c_{123} & k_6 c_{123} \end{bmatrix} \quad (170)$$

where

$$\begin{aligned} k_4 &= b_0 \\ k_5 &= a_1 + b_1 \\ k_6 &= a_2 + b_2 \end{aligned} \quad (171)$$

Comparing J_{bac} and J_{3lk} , it appears that the only effect of the translating base is to change the coefficients of the sine and cosine functions in the elements of the Jacobian. Essentially, the geometric link lengths of J_{3lk} are replaced with “generalized link lengths” created by combinations of lengths and masses in J_{bac} . The joint angles that comprise a singular configuration for the base-attitude controlled SMM are identical to those of the fixed-base three-link arm.

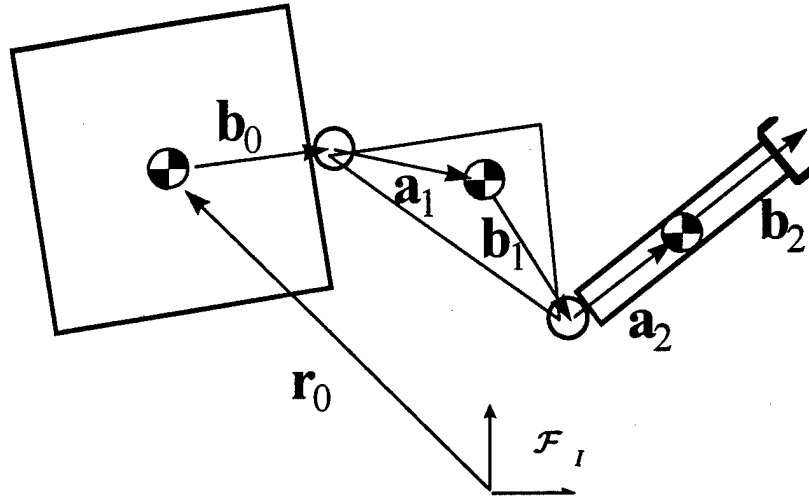


Figure 15. Two-Link SMM With First-Link Center of Mass Offset

This is not necessarily always the case. The example above assumed that the centers of mass of the links lay on a line connecting the joints. If this is not the case, the angles which create a singularity of the SMM may be different from the singular angles of the fixed-base counterpart.

Consider a modification of the example above, in which the center of mass of the first link does not lie on a line connecting the joints, as shown in Figure 15. The end-effector position is given vectorially by Eq. (143), just as before. Now the end-effector velocity can be written in vector form as

$$\dot{\mathbf{r}} = \mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{b}_0 + \boldsymbol{\omega}_1 \times (\mathbf{a}_1 + \mathbf{b}_1) + \boldsymbol{\omega}_2 \times (\mathbf{a}_2 + \mathbf{b}_2) \quad (172)$$

where $\boldsymbol{\omega}_i$ is the angular velocity of the i^{th} body with respect to the inertial frame. This can be rewritten as

$$\begin{aligned} \dot{\mathbf{r}} = & \mathbf{v}_0 + \boldsymbol{\omega}_0 \times (\mathbf{b}_0 + \mathbf{a}_1 + \mathbf{b}_1 + \mathbf{a}_2 + \mathbf{b}_2) \\ & + \boldsymbol{\omega}_1^1 \times (\mathbf{a}_1 + \mathbf{b}_1 + \mathbf{a}_2 + \mathbf{b}_2) + \boldsymbol{\omega}_1^2 \times (\mathbf{a}_2 + \mathbf{b}_2) \end{aligned} \quad (173)$$

where ω_i^j is the angular velocity of the j^{th} -body with respect to frame \mathcal{F}_i . The momentum of the system is

$$\begin{aligned}\mathbf{p} &= 0 = m_0\mathbf{v}_0 + m_1\mathbf{v}_1 + m_2\mathbf{v}_2 \\ &= m_T\mathbf{v}_0 + \omega_0 \times [m_1(\mathbf{b}_0 + \mathbf{a}_1) + m_2(\mathbf{b}_0 + \mathbf{a}_1 + \mathbf{b}_1 + \mathbf{a}_2)] \\ &\quad + \omega_0^1 \times [m_1\mathbf{a}_1 + m_2(\mathbf{a}_1 + \mathbf{b}_1 + \mathbf{a}_2)] + \omega_1^2 \times m_2\mathbf{a}_2\end{aligned}\tag{174}$$

Solving this for \mathbf{v}_0 and substituting into Eq. (173) gives

$$\begin{aligned}\dot{\mathbf{r}} &= \frac{1}{m_T} \{ \omega_0 \times [m_0(\mathbf{b}_0 + \mathbf{a}_1) + (m_0 + m_1)(\mathbf{b}_1 + \mathbf{a}_2) + m_T\mathbf{b}_2] \\ &\quad + \omega_0^1 \times [m_0\mathbf{a}_1 + (m_0 + m_1)(\mathbf{b}_1 + \mathbf{a}_2) + m_T\mathbf{b}_2] \\ &\quad + \omega_1^2 \times [(m_0 + m_1)\mathbf{a}_2 + m_T\mathbf{b}_2] \}\end{aligned}\tag{175}$$

For the planar SMM, all of the angular velocity vectors are perpendicular to the plane of motion, so the configuration is singular only if all three of the vectors “crossed by” an angular velocity vector are collinear. To find the angles (θ_1, θ_2) that result in this singular configuration, a geometric approach can be used. Define three vectors \mathbf{u}_1 , \mathbf{u}_2 , and \mathbf{u}_3 as

$$\begin{aligned}\mathbf{u}_1 &= m_0(\mathbf{b}_0 + \mathbf{a}_1) + (m_0 + m_1)(\mathbf{b}_1 + \mathbf{a}_2) + m_T\mathbf{b}_2 \\ \mathbf{u}_2 &= m_0\mathbf{a}_1 + (m_0 + m_1)(\mathbf{b}_1 + \mathbf{a}_2) + m_T\mathbf{b}_2 \\ \mathbf{u}_3 &= (m_0 + m_1)\mathbf{a}_2 + m_T\mathbf{b}_2\end{aligned}\tag{176}$$

The angle between the first and second links is θ_2 , and by definition is the angle between the vectors $\mathbf{a}_1 + \mathbf{b}_1$ and $\mathbf{a}_2 + \mathbf{b}_2$. To determine this angle, when the system is singular, start by assuming \mathbf{u}_2 is collinear to \mathbf{u}_3 . Then \mathbf{u}_2 is a scalar multiple of \mathbf{u}_3 , and we can write

$$\mathbf{u}_2 = k\mathbf{u}_3\tag{177}$$

$$m_0\mathbf{a}_1 + (m_0 + m_1)(\mathbf{b}_1 + \mathbf{a}_2) + m_T\mathbf{b}_2 = k((m_0 + m_1)\mathbf{a}_2 + m_T\mathbf{b}_2)\tag{178}$$

$$m_0\mathbf{a}_1 + (m_0 + m_1)\mathbf{b}_1 = (k - 1)((m_0 + m_1)\mathbf{a}_2 + m_T\mathbf{b}_2)\tag{179}$$

We have assumed that the center of mass of link two does lie on the line connecting joint two and the end-effector (see Figure 15), so vectors \mathbf{a}_2 , \mathbf{b}_2 , and $\mathbf{a}_2 + \mathbf{b}_2$ share a common direction. Then

$$m_0 \mathbf{a}_1 + (m_0 + m_1) \mathbf{b}_1 = k' \mathbf{a}_2 = k'' (\mathbf{a}_2 + \mathbf{b}_2) \quad (180)$$

for some scalars k' and k'' . This equation demonstrates that the vector $m_0 \mathbf{a}_1 + (m_0 + m_1) \mathbf{b}_1$ is collinear with the vector $\mathbf{a}_2 + \mathbf{b}_2$. Then the angle between the vector $m_0 \mathbf{a}_1 + (m_0 + m_1) \mathbf{b}_1$ and the vector $\mathbf{a}_1 + \mathbf{b}_1$ is equal to $\pm \theta_2$ if $k'' > 0$ or $\pi \pm \theta_2$ if $k'' < 0$. This angle is easily found using the definition of the dot product,

$$\mathbf{u} \cdot \mathbf{v} \triangleq \|\mathbf{u}\| \|\mathbf{v}\| \cos \theta \quad (181)$$

Applying this to find θ_2 results in the equation

$$\pm \cos \theta_2 = \frac{(\mathbf{a}_1 + \mathbf{b}_1) \cdot (m_0 \mathbf{a}_1 + (m_0 + m_1) \mathbf{b}_1)}{\|\mathbf{a}_1 + \mathbf{b}_1\| \|m_0 \mathbf{a}_1 + (m_0 + m_1) \mathbf{b}_1\|} \quad (182)$$

This equation has at most two solutions on the interval $0 \leq \theta_2 \leq 2\pi$, for each choice of sign on the left hand side for a total of four possible solutions. Only two of these solutions are consistent with the definition of θ_2 (positive clockwise, so that $(\mathbf{a}_1 + \mathbf{b}_1) \times (\mathbf{a}_2 + \mathbf{b}_2)$ is always in a positive z -direction). So for a given SMM, there are only two possible singular values of θ_2 .

In the same way, the collinearity of vectors \mathbf{u}_1 and \mathbf{u}_2 can be used to determine the angles of θ_1 which result in a singular configuration. In θ_1 - θ_2 space, there is a total of four singular points for the SMM on the interval $0 \leq \theta_1 \leq 2\pi$ and $0 \leq \theta_2 \leq 2\pi$.

This derivation has two notable results. First, an offset center of mass, coupled with the free translation of the base, can move the singular configuration away from the configuration suggested by the kinematic structure of the arm. Second, and perhaps most important, is that although the singular configuration may not be identical to that of the corresponding fixed-base robot, the singularities are still discrete points in joint space. By controlling the base attitude, the singularities are distinct configurations of the arm, unlike the dynamic singularities seen for a free-floating base which form curves in θ_1 - θ_2 space. Translated to inertial workspace, this means that the base atti-

tude controlled planar SMM, even with offset centers of mass, has only one-dimensional regions of singularity, whereas the free-floating SMM has the two-dimensional singularity regions or Path Dependent Workspace.

4.7 Summary

We began this chapter with a review of some of the fundamental concepts in the control of free-floating SMMs. These included Umetani and Yoshida's Generalized Jacobian Matrix (GJM) and Papadopolous' definition of dynamic singularities. The independence of dynamic singularities from the spacecraft position or attitude was noted and we discussed how this allows a mapping of singular joint configurations to a region of reachable work space known as the Path Dependent Workspace (PDW). The relation between singularities and the ratio of base inertia to arm inertia was mentioned, giving rise to the idea of dynamic singularity manifolds as perturbations of kinematic singularity manifolds.

In the subsequent sections of the chapter, we considered some new free-floating design alternatives for alleviating the problems associated with dynamic singularities. The effect of redundancy was demonstrated, where we showed that redundancy increases the PDW, but also enables alternate joint trajectories that may help avoid singularities. Prismatic joints were shown to be effective in eliminating singularity sets, decreasing the size of the PDW. Joint limits were shown to reduce the size of the PDW as well. A combination of redundancy and prismatic joints was shown to eliminate dynamic singularities completely for a planar case.

The last method of eliminating dynamic singularities considered was not an alternative free-floating design, but the approach of adding base control. We showed that, in general, complete base control totally eliminates dynamic singularities. For base attitude control, examples were offered that indicate singularities are equivalent to kinematic singularities, in the sense that their Path De-

pendent Workspace reduces to one dimension in the planar case and two dimensions in the spatial case.

Chapter 5 - SMM Control

In Chapter 4, several SMM actuation concepts were introduced, momentum-constrained Jacobians were developed, and characteristics of dynamic singularities were investigated. In this chapter, these ideas are incorporated into the development of a new SMM controller. Stability analysis of the controller confirms the importance of avoiding dynamic singularities. Comparing the use of momentum-constrained Jacobians and the standard manipulator Jacobian in the controller suggests the primary advantage of using momentum-constrained Jacobians is significantly finer control over the end-effector trajectory. The effectiveness of base-attitude control in avoiding singularity problems is demonstrated. Finally, a reduced base-torque controller is developed which combines the favorable aspects of both the free-floating and base-attitude controlled approaches.

The SMM control problem, as considered here, is to choose appropriate joint torques (and base torques/forces where base actuation is used) to steer the end-effector from one position to another, preferably along a prescribed path. The input and output are in fundamentally different spaces, joint space and workspace. In the control of terrestrial robots, this division has led to two primary approaches to control, one based on inverse kinematic relations, and the other on the manipulator Jacobian.

In the first approach, the control problem is divided into two subproblems, path planning and joint control. Path planning consists primarily of determining the joint trajectory that will result in the desired end-effector trajectory using an inverse kinematic solution. The joint controller then determines the torques which will provide this joint trajectory. Many effective and general methods of joint control exist. These range from simple independent feedback loops for each joint, which treat nonlinear dynamic effects as disturbances, to nonlinear methods like the method of computed torques, which uses a dynamic model of the system to improve the tracking. In contrast, general

solutions to the inverse kinematics problem do not exist, since solutions are dependent on the particular robot geometry. Instead, a pool of simplifying techniques have accumulated which attempt to divide the problem into smaller problems, which can be solved and used to construct a complete solution to the original problem.

The “inverse kinematics plus joint control” approach can be applied to SMM control, but several complications arise. The end-effector position no longer depends only on the joint angles of the manipulator, but also on the position and orientation of the base. For a spatial manipulator, this adds six DOF to the already difficult inverse kinematics problem. The increased complexity makes any controller requiring an inverse kinematic solution somewhat unattractive. Furthermore, if the SMM is free-floating, the joint control portion of this control approach is also considerably more difficult since there is no direct control over the base position or orientation, so traditional joint control methods are unusable. A solution to this new joint control problem does exist (recall Reyhanoglu and McClamroch [39]), but this method is not well suited for a trajectory-following problem.

The alternative control approach, based on the manipulator Jacobian, entirely avoids the problem of solving the inverse kinematics, and does not require control of the unactuated degrees of freedom. The main feedback is accomplished using workspace variables, with end-effector velocity or acceleration commands being generated based on end-effector position and/or velocity errors. These commands are converted to joint space commands using the Jacobian inverse. This type of method was first proposed by Whitney [50] using velocities, and is known as Resolved Motion Rate Control. It was extended to the acceleration level by Luh *et al.* [22], where it is referred to as Resolved Acceleration Control. The greatest drawback of controllers based on this approach is that they can fail completely at singularities of the Jacobian. Despite this, it can be a powerful approach, and can be extended to SMM control. This is the approach used in this work.

5.1 Basic Controller

In this section, a new SMM control scheme is developed. This controller, termed the “basic SMM controller,” provides a framework for fair comparison of a variety of SMM control concepts. The controller consists of an outer feedback loop to provide end-effector position control and an inner feedback loop for joint velocity control. Each SMM control concept creates a variation of the basic controller according to the method used to convert work space velocities to joint space velocities. The framework extends to both free-floating concepts and base controlled concepts by defining joint space to include all actuated degrees of freedom in the system. The transformation from work space velocity to joint space velocity implies an inversion of a Jacobian, but no particular inversion method or Jacobian type is assumed in the basic SMM controller.

The outer loop is based on Resolved Motion Rate Control [50], using proportional feedback of end-effector position error and the desired end-effector velocity to produce an end-effector velocity command. This workspace command velocity is converted to a joint velocity command using a Jacobian inverse. The inner loop then drives the system to the desired joint velocities. Since the inverse kinematic problem is not solved in this scheme, the final joint position is unknown, so only the joint velocity is controlled. The joint control problem is highly nonlinear, and the inner loop control law is derived so as to guarantee the stability of the inner loop by Lyapunov’s direct method [47]. The inner loop is based on the method of computed torques [27, 42], modified to control joint velocity instead of joint position. A simplification of the control law which reduces computational costs is also considered.

5.1.1 Controller Derivation

To derive the outer loop controller, we begin by defining the position error as

$$e_1 = r_d - r \tag{183}$$

where r is the current position of the end-effector in inertial space and r_d is the desired end-effector position. The error rate is then

$$\dot{e}_1 = \dot{r}_d - \dot{r} \quad (184)$$

Feedback is synthesized using the relation,

$$\dot{e}_1 = -K_1 e_1 \quad (185)$$

where K_1 is a positive definite matrix chosen to make the system stable and give the desired response time. Substituting the definitions of e_1 and \dot{e}_1 , this equation becomes

$$\dot{r}_d - \dot{r} = -K_1 (r_d - r) \quad (186)$$

$$\dot{r} = \dot{r}_d + K_1 (r_d - r) \quad (187)$$

The end-effector velocity given on the left-hand side of Eq. (187) is used as the command input, (*i.e.*, let $\dot{r}_c = \dot{r}$), to the workspace controller. If the end-effector velocity command can be achieved instantaneously, this control law guarantees that the system will converge asymptotically to the desired end-effector position.

In reality, the end-effector velocity command, \dot{r}_c , must be converted to a joint velocity command, $\dot{\theta}_c$. This can be done by inverting the manipulator Jacobian,

$$\dot{\theta}_c = J^{-1} \dot{r}_c \quad (188)$$

Two different forms of the manipulator Jacobian could be used in Eq. (188). The standard fixed-base manipulator Jacobian is the most straightforward choice. However, it does not accurately reflect the true relationship between end-effector velocity and joint velocity, failing to account for the moving base. A momentum-constrained Jacobian, such as Umetani and Yoshida's GJM for the free-floating case, or Wee and Walker's dynamic Jacobian for the base attitude controlled case, provides a more precise solution. The effect of this choice on the SMM response is investigated in the following sections.

The inner loop of the basic SMM controller is the joint space controller, which chooses torques to drive the system to the commanded joint velocities. This is a highly nonlinear problem. To derive a suitable controller, we use the approach of writing a Lyapunov function and choosing a control law which guarantees asymptotic stability.

The equations of motion can be written as

$$\tilde{M}\ddot{\theta} + \tilde{C}\dot{\theta} = \tau \quad (189)$$

where the unactuated coordinates from the base have been eliminated using the momentum equations (see Appendix A). Define velocity and acceleration errors

$$e_2 = \dot{\theta} - \dot{\theta}_c \quad (190)$$

$$\dot{e}_2 = \ddot{\theta} - \ddot{\theta}_c \quad (191)$$

Define a Lyapunov function

$$V = e_2^\top \tilde{M} e_2 \quad (192)$$

Note that $V > 0$ for all θ and $\dot{\theta}$. Differentiating Eq. (192) gives

$$\dot{V} = 2e_2^\top \tilde{M} \dot{e}_2 + e_2^\top \dot{\tilde{M}} e_2 \quad (193)$$

Using Eqs. (189) and (191), and assuming $\ddot{\theta}_c \equiv 0^9$, Eq. (193) becomes

$$\dot{V} = 2e_2^\top (\tau - \tilde{C}\dot{\theta}) + e_2^\top \dot{\tilde{M}} e_2 \quad (194)$$

Now we can choose the control law

$$\tau = -K_2 e_2 + \tilde{C}\dot{\theta}_c \quad (195)$$

so that Eq. (194) becomes

$$\dot{V} = -2e_2^\top K_2 e_2 + e_2^\top (\dot{\tilde{M}} - 2\tilde{C}) e_2 \quad (196)$$

⁹In the development of the SMM equations of motion, effects related to the structural flexibility of the links were neglected. Since spacecraft will generally be lightweight structures, this assumption necessarily implies that commanded motions will be slow and trajectories will be planned so as to have negligible acceleration. Therefore, it is assumed that $\ddot{\theta}_c \equiv 0$. This allows a stabilizing control law using only a feedforward velocity term. If the assumption is not made, the control law would also require a feedforward acceleration term. To compute this term, the derivative of the Jacobian is needed, creating a significantly higher computational load on the controller.

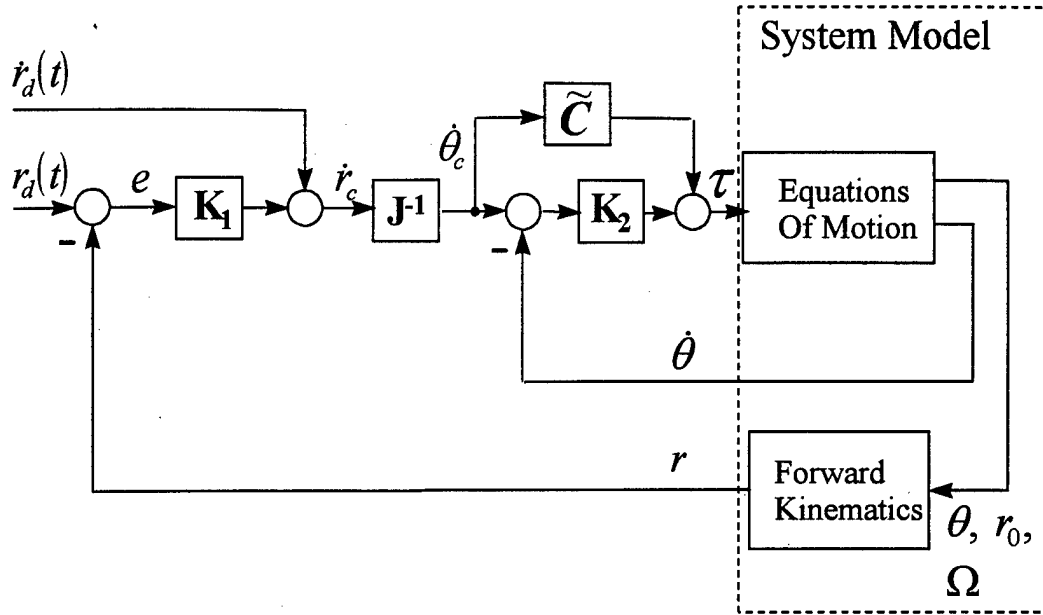


Figure 16. Basic Nonlinear SMM Controller

The second term on the right-hand side of Eq. (196) can be shown to be zero for all $\theta, \dot{\theta}$ (see Appendix B.3), so according to the direct method of Lyapunov, choosing a positive definite matrix K_2 guarantees that system will asymptotically converge to the desired velocity.

Figure 16 shows a block diagram of the basic SMM controller. The block labeled “ J^{-1} ” is intended to represent whatever method of converting from workspace velocity to joint space velocity is to be used in a particular variation. The “System Model” represents the actual dynamics of the system. For a real system, torques (τ) would be the inputs and joint velocity ($\dot{\theta}$) and end-effector position (r) would be measured values. For simulation purposes, the block labeled “Equations of Motion” represents the numerical integration of state Eqs. (95) and (96), where \tilde{M} , \tilde{C} , and τ are used in place of M , C , and Q , respectively. Finally, the block labeled “Forward Kinematics” represents the forward kinematic equation, Eq. (5). Specific elements of the unreduced equations of motion matrices M and C are given for the planar two-link case in Appendix C. The reduced matrices \tilde{M} and \tilde{C} are computed numerically using the technique shown in Appendix B.2.

5.1.2 A Linearization Of The Control Law

A disadvantage of the control law given by Eq. (195) is the significant computational cost of determining \tilde{C} at every step. As an alternative, consider the effect of using a constant average value for \tilde{C} . This would change the control law to the linear form

$$\tau = -K_2 e_2 + \tilde{C}_{avg} \dot{\theta}_c \quad (197)$$

so that the derivative of the Lyapunov function becomes

$$\dot{V} = -2e_2^\top K_2 e_2 + e_2^\top (\dot{\tilde{M}} - 2\dot{\tilde{C}}) e_2 + e_2^\top (\tilde{C}_{avg} - \tilde{C}) \dot{\theta}_c \quad (198)$$

$$= -2e_2^\top K_2 e_2 + e_2^\top (\tilde{C}_{avg} - \tilde{C}) \dot{\theta}_c \quad (199)$$

If \tilde{C}_{avg} represents the center value of \tilde{C} for a typical task, then the second term should be small. By choosing a large value for K_2 , the first term of Eq. (199) will dominate the left-hand side, and \dot{V} will remain negative, assuring the stability of the inner loop. Assuming that there is not a preferred manipulator configuration, a reasonable first choice is to let $\tilde{C}_{avg} = 0$. The resulting control law is

$$\tau = -K_2 e_2 \quad (200)$$

This is also the stabilizing control law that would result if we had assumed all the nonlinear velocity terms were negligible. In practice, these will probably not be completely negligible. However, given our concern for avoiding flexible effects it is not unrealistic to expect low velocities and Eq. (199) indicates that destabilizing effects of the nonlinear velocity terms can be reduced by increasing the inner loop gain (K_2). Based on our simulation experience, the control law given in Eq. (200) was a satisfactory choice, but extensive testing over the expected range of operation would need to be done before using the linearized controller in a real system.

Figure 17 shows a block diagram of the basic SMM controller using the control law in Eq. (200), so no feedforward path is shown. This is the form of the basic controller that was used in the examples shown in the remainder of this work. The conclusions drawn about the effectiveness of

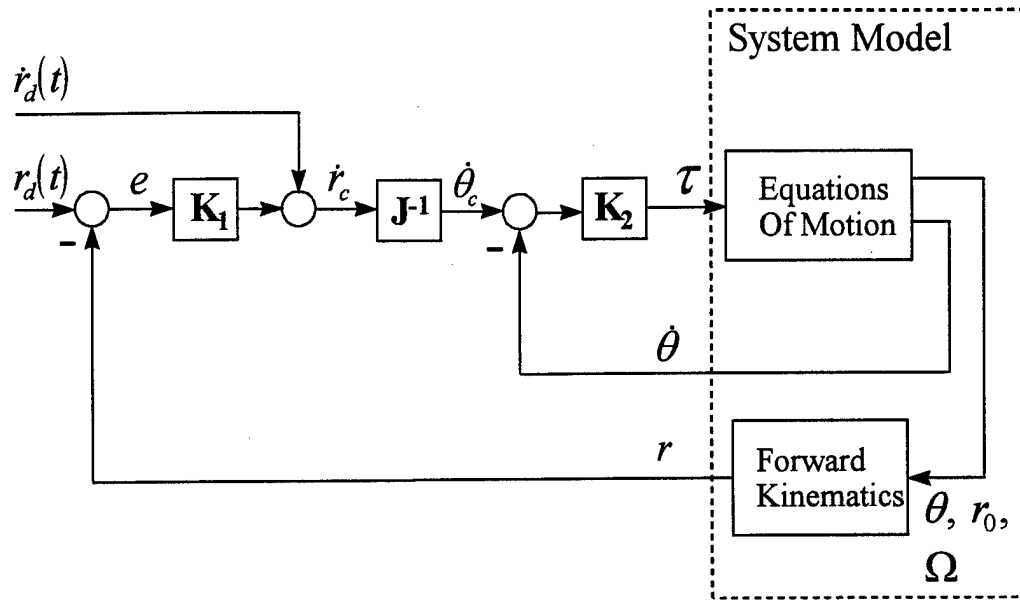


Figure 17. Basic SMM Controller

various levels of base control remain valid for the nonlinear control law, since it's use only further enhances the stability of the system.

5.1.3 Stability of the Basic SMM Controller

The SMM controller derived above consisted of two feedback loops—an outer workspace loop and an inner joint space loop—connected by a Jacobian inverse. Each loop was designed to be stable independently, but this does not guarantee the stability of the complete system. Therefore, it is important to analyze the complete system to find conditions under which stability is guaranteed.

Consider the Lyapunov function,

$$V = \frac{1}{2}e_1^T K_1 e_1 + \frac{1}{2}e_2^T \tilde{M} e_2 \quad (201)$$

where $e_1 = r - r_d$ and $e_2 = \dot{\theta} - \dot{\theta}_c$. Differentiating Eq. (201) gives

$$\dot{V} = e_1^\top K_1 \dot{e}_1 + e_2^\top \tilde{M} \dot{e}_2 + \frac{1}{2} e_2^\top \dot{\tilde{M}} e_2 \quad (202)$$

where $\dot{e}_1 = \dot{r}$ and $\dot{e}_2 = \ddot{\theta}$. Using the equations of motion from Eq. (189) and the control law from Eq. (195), Eq. (202) becomes

$$\dot{V} = e_1^\top K_1 \dot{r} - e_2^\top (K_2 + \tilde{C}) e_2 + \frac{1}{2} e_2^\top \dot{\tilde{M}} e_2 \quad (203)$$

or

$$\dot{V} = e_1^\top K_1 \dot{r} - e_2^\top K_2 e_2 + \frac{1}{2} e_2^\top (\dot{\tilde{M}} - 2\tilde{C}) e_2 \quad (204)$$

As in Eq. (196), the last term is zero, so

$$\dot{V} = e_1^\top K_1 \dot{r} - e_2^\top K_2 e_2 \quad (205)$$

We have already acknowledged the failure of this controller when the manipulator Jacobian becomes singular, so nothing is lost by assuming here that the Jacobian is nonsingular. Furthermore, assume for now that the Jacobian is square. Then the Lyapunov function derivative can be written as a quadratic form of the workspace position error, e_1 , and the joint velocity, $\dot{\theta}$. The end-effector velocity, \dot{r} , and joint velocity error, e_2 , are transformed appropriately using the Jacobian and its inverse,¹⁰

$$\dot{r} = J\dot{\theta} \quad (206)$$

$$e_2 = \dot{\theta} - \dot{\theta}_c = \dot{\theta} - J^{-1}\dot{r}_c \quad (207)$$

From Figure 17, $\dot{r}_c = -K_1 e_1$, so that

$$e_2 = \dot{\theta} + J^{-1} K_1 e_1 \quad (208)$$

and

$$\begin{aligned} \dot{V} = & e_1^\top K_1 J \dot{\theta} - \dot{\theta}^\top K_2 \dot{\theta} - e_1^\top K_1 J^{-\top} K_2 J^{-1} K_1 e_1 \\ & - \dot{\theta}^\top K_2 J^{-1} K_1 e_1 - e_1^\top K_1 J^{-\top} K_2 \dot{\theta} \end{aligned} \quad (209)$$

¹⁰Recognize that the equation $\dot{r} = J\dot{\theta}$ is only exact if a momentum constrained Jacobian is used, and the column matrix $\dot{\theta}$ represents all actuated coordinates. In a case where the standard manipulator Jacobian is used in generating $\dot{\theta}_c$, stability is not proven, it is merely suggested.

where $J^{-\top} = (J^{-1})^\top$. Equation (209) can be written in matrix form as,

$$\dot{V} = - \begin{bmatrix} e_1^\top K_1 & \dot{\theta}^\top \end{bmatrix} \begin{bmatrix} J^{-\top} K_2 J^{-1} & \frac{1}{2} J - J^{-\top} K_2 \\ \frac{1}{2} J^\top - K_2 J^{-1} & K_2 \end{bmatrix} \begin{bmatrix} K_1 e_1 \\ \dot{\theta} \end{bmatrix} \quad (210)$$

$$= -x^\top B x \quad (211)$$

The definiteness of the symmetric matrix B in the middle of the right hand side of Eq. (210) is the key to the stability of the controller. The eigenvalues of B are the values λ for which the equation

$$(B - \lambda U) x = 0 \quad (212)$$

has a solution for some nonzero x (the eigenvectors). Substituting the definition of B into the equation above gives

$$\begin{bmatrix} J^{-\top} K_2 J^{-1} - \lambda_1 U & \frac{1}{2} J - J^{-\top} K_2 \\ \frac{1}{2} J^\top - K_2 J^{-1} & K_2 - \lambda_2 U \end{bmatrix} x = 0 \quad (213)$$

where the set of eigenvalues, λ , is divided into two subsets, λ_1 and λ_2 . Using elementary matrix operations this linear system of equations can be rewritten as

$$\begin{bmatrix} 2J^{-\top} K_2 J^{-1} - \frac{1}{2} U - \lambda_1 U & \frac{1}{2} (U - 2A) \\ 0 & \frac{1}{2} J^\top J - \lambda_2 U \end{bmatrix} x = 0 \quad (214)$$

Equation (214) is in block diagonal form and the eigenvalues of B are the eigenvalues of the blocks on the diagonal. Since J is assumed nonsingular, the form of the lower diagonal block, $J^\top J$, is sufficient to guarantee that it is positive definite, so all the elements of λ_2 are positive. The upper diagonal block, $2J^{-\top} K_2 J^{-1} - \frac{1}{2} U$, can be made positive definite by choosing the appropriate gains. First, choose the K_2 gain matrix to be a scalar multiple of the identity matrix, $K_2 = kU$, with $k > 0$. Then consider the eigenvalue equation for the block,

$$\left(2kJ^{-\top} J^{-1} - \frac{1}{2} U - \lambda U \right) x = 0 \quad (215)$$

$$kJ^{-\top} J^{-1} x = \left(\frac{1}{4} + \frac{\lambda_1}{2} \right) x \quad (216)$$

Now the form of $J^{-\top} J^{-1}$ and the nonsingularity of J ensure that all of its eigenvalues are positive.

Suppose the smallest is λ_{\min} . Then the smallest eigenvalue of $kJ^{-\top} J^{-1}$ is $k\lambda_{\min}$. Then from Eq.

(216),

$$k\lambda_{\min} = \frac{1}{4} + \frac{\min(\lambda_1)}{2} \quad (217)$$

Now to guarantee that the $\min(\lambda_1) > 0$, choose k such that

$$\min(\lambda_1) = 2k\lambda_{\min} - \frac{1}{2} > 0 \quad (218)$$

$$k > \frac{1}{4\lambda_{\min}} \quad (219)$$

Therefore, matrix B in Eq. (210) must be positive definite for large enough value of k .

Therefore, the basic SMM controller guarantees stability and convergence to the desired end-effector position provided:

- K_2 is chosen as kU , with k chosen sufficiently large positive. Essentially, this means that the inner loop must be fast enough to make the nonlinear effects on the outer loop negligible.
- The manipulator Jacobian, J , remains nonsingular. This underscores the importance of avoiding dynamic singularities.

5.1.4 Stability in the Redundant Case

In some cases, the Jacobian relating the end-effector velocity and the actuated coordinate velocities will not be square. The cases in which there are more actuated coordinates than end-effector coordinates are considered *redundant*, and the Jacobian will have more columns than rows. The actuator command will generally consist of a term containing the pseudoinverse of the Jacobian multiplied by the workspace command and another term indicating the desired *null motion*, joint velocity combinations which are in the nullspace of the Jacobian. Then the command velocity appears in the form

$$\dot{\theta}_c = J^\# \dot{r}_c + Nz \quad (220)$$

where the matrix N maps a vector z into the null space of the Jacobian. Therefore the null term has the property that $JNz = 0$. If the Jacobian has full row rank, it is considered nonsingular and the additional property $JJ^\# = U$ applies.

The stability proof in the previous section can be done for the redundant case as well. The proof is identical through Eq. (205). Then

$$\dot{r} = J\dot{\theta} \quad (221)$$

$$e_2 = \dot{\theta} - \dot{\theta}_c = \dot{\theta} - J^\# \dot{r}_c - Nz \quad (222)$$

Assume that the null term is dependent on the end-effector command and can be written $Nz = \hat{N}\dot{r}_c$.

Then from Figure 17, $\dot{r}_c = -K_1 e_1$, so that

$$e_2 = \dot{\theta} + (J^\# + \hat{N}) K_1 e_1 \quad (223)$$

and

$$\begin{aligned} \dot{V} = & e_1^\top K_1 J \dot{\theta} - \dot{\theta}^\top K_2 \dot{\theta} - e_1^\top K_1 (J^\# + \hat{N})^\top K_2 (J^\# + \hat{N}) K_1 e_1 \\ & - \dot{\theta}^\top K_2 (J^\# + \hat{N}) K_1 e_1 - e_1^\top K_1 (J^\# + \hat{N})^\top K_2 \dot{\theta} \end{aligned} \quad (224)$$

This becomes

$$\dot{V} = - \begin{bmatrix} e_1^\top K_1 & \dot{\theta}^\top \end{bmatrix} \begin{bmatrix} (J^\# + \hat{N})^\top K_2 (J^\# + \hat{N}) & \frac{1}{2}J - (J^\# + \hat{N})^\top K_2 \\ \frac{1}{2}J^\top - K_2 (J^\# + \hat{N}) & K_2 \end{bmatrix} \begin{bmatrix} K_1 e_1 \\ \dot{\theta} \end{bmatrix} \quad (225)$$

As before, the definiteness of the symmetric matrix in the middle of the right hand side is the key

to the stability of the system. The eigenvalues are found by solving

$$\begin{bmatrix} (J^\# + \hat{N})^\top K_2 (J^\# + \hat{N}) - \lambda_1 U & \frac{1}{2}J - (J^\# + \hat{N})^\top K_2 \\ \frac{1}{2}J^\top - K_2 (J^\# + \hat{N}) & K_2 - \lambda_2 U \end{bmatrix} x = 0 \quad (226)$$

Using elementary matrix operations this linear system of equations can be rewritten as

$$\begin{bmatrix} 2(J^\# + \hat{N})^\top K_2 (J^\# + \hat{N}) - \frac{1}{2}U - \lambda_1 U & \frac{1}{2}J - (J^\# + \hat{N})^\top K_2 \\ 0 & \frac{1}{2}J^\top J - \lambda_2 U \end{bmatrix} x = 0 \quad (227)$$

Equation (227) is in block diagonal form and the eigenvalues are the eigenvalues of the blocks on the diagonal. Since J is assumed nonsingular, the form of the lower diagonal block, $J^\top J$, is sufficient

to guarantee that it is positive definite even for the nonsquare J (J is $m \times n$ with $m < n$), so all the elements of λ_2 are positive. The upper diagonal block, $2 \left(J^\# + \hat{N} \right)^\top K_2 \left(J^\# + \hat{N} \right) - \frac{1}{2}U$, can be made positive definite by choosing the appropriate gains in exactly the same way as for the square case. This is guaranteed because $J^\# + \hat{N}$ has independent columns and full column rank (the columns of $J^\#$ spans the range of J and the columns of \hat{N} are in the null space of J) so $\left(J^\# + \hat{N} \right)^\top \left(J^\# + \hat{N} \right)$ is positive definite just as $J^{-\top} J^{-1}$ was positive definite for square (nonsingular) J .

Therefore, the redundant case is stable under the same conditions as the nonredundant case: the inner loop gain must be sufficiently large and the Jacobian must be nonsingular.

5.1.5 A Note on Gain Selection

To implement the SMM controller, appropriate values were needed for the gain matrices, K_1 and K_2 . By choosing these matrices to be scalar multiples of the identity matrix, the system response was tuned with two values. The outer loop gain, K_1 , was the primary tool for adjusting the response time. The inner loop gain, K_2 , was sized to guarantee stability of the overall system and produce a desirable “linear” type of end-effector response. Using a moderate value for K_2 was generally sufficient for stability, but often resulted in unintuitive nonlinear response. Using a high value for K_2 not only ensured stability, but also increased the speed of the inner loop to a point where the system behaved as if the linear outer loop was the true model of the dynamics.

Ideally, we wanted the end-effector to mimic a simple linear system, so we used a high K_2 value. Experience showed that to get close to the desired linear system behavior required that K_2 be at least two orders of magnitude greater than K_1 . For our simulations, increasing K_2 by much more than two orders of magnitude significantly increased the numerical integration time required for a typical maneuver. In a real system, the upper limit for K_2 would be determined by practical

considerations. The controller would be implemented digitally, and as K_2 increased, the necessary sampling rate would increase. The upper limit of K_2 would then depend upon the measurement rates and the computational speed of the controller.

5.2 Free-Floating Case

This section compares the performance of a free-floating SMM using variations on the basic SMM controller presented in the previous section. Recalling that the purpose of the controller is to steer the end-effector from one position to another along a prescribed path, performance is judged based on three metrics. The first is the speed with which the desired position is reached and maintained, often referred to as the settling time. This provides a measure of the maneuver quality, since generally there will be some limit to the time allowed for the maneuver. At the least, a successful maneuver must have a finite settling time. The second metric is the actuator torque requirements during the maneuver, expressed as the magnitude of the torque vector. This is a measure of the maneuver cost. Often, the integral of torque vector magnitude is most important, especially for those torques which must be generated by thrusters. In this case, the metric is directly proportional to the total fuel expended. The peak torques during the maneuver can also be of interest, since they will affect the size of actuators required. The final performance indicator is the end-effector path during the maneuver. Since following a prescribed path can be essential to avoiding obstacles in workspace, the extent to which the end-effector deviates from this path is another important maneuver quality measure. The magnitude of the path error can be tracked throughout the maneuver and the integral of this error provides a convenient number for comparison.

Formulae for the metrics are

$$\text{TimeMetric} = \int dt \quad (228)$$

$$\text{TorqueMetric} = \int \|\tau\| dt \quad (229)$$

$$\text{PathMetric} = \int \|r_d(t) - r(t)\| dt \quad (230)$$

where in each case the integration limits are from the maneuver start time to the maneuver end time.

For the free-floating case, we considered three variations of the basic SMM controller. The first is the naive approach, using the inverse of the standard manipulator Jacobian to determine the joint velocity commands from the end-effector velocity command (*i.e.*, “ J^{-1} ” in Figure 17 indicates J_{fb}^{-1}). The second approach is to use the standard manipulator Jacobian, but modify the end-effector velocity command with base motion feedback. The last variation is to use the GJM to relate joint velocities and end-effector velocity (*i.e.*, “ J^{-1} ” in Figure 17 indicates \tilde{J}^{-1}).

The first option, using only the standard manipulator Jacobian, completely ignores the effect of the base movement induced by the moving arm. It will converge to the desired position as long as the manipulator Jacobian remains nonsingular. For tasks where point-to-point movement of the end-effector is the only concern, this controller may be sufficient. However, it generally results in significant deviations from the desired end-effector path, so it is unsuitable for tasks where precise path following is important. This option does have a few advantages. It is conceptually straightforward, relatively low in terms of computational costs, and requires no information on the system mass properties.

The second option is to retain the standard manipulator Jacobian, but feed back base velocity measurements. This method, known as reaction compensation control, was suggested by Spofford and Akin [41]. Using this method, the base velocity is measured and converted to an end-effector velocity using the forward velocity kinematic relations. The end-effector velocity command, \dot{r}_c , is formed from the difference of the end-effector velocity resulting from the base motion and the command produced by Eq. (187). The joint velocity commands are then generated using the standard manipulator Jacobian. This method requires a more significant modification to the basic SMM controller than other methods. The block diagram in Figure 18 shows the addition of a third feedback

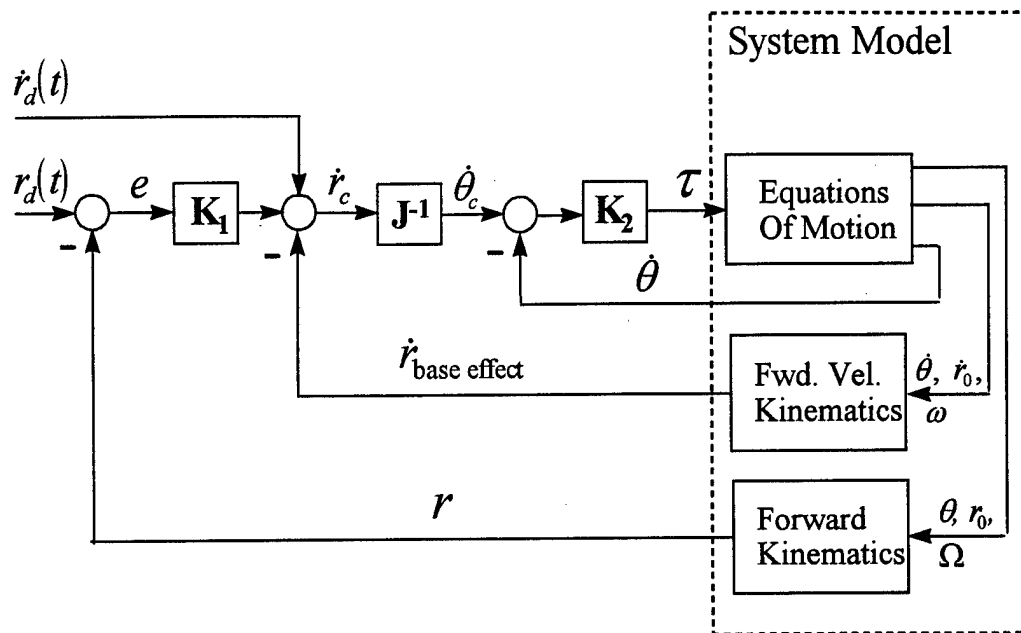


Figure 18. SMM Controller with Base-Motion Feedback

loop. This method has advantages similar to the first option and improves the tracking of the desired path. The reaction compensation controller variation also may fail if the manipulator Jacobian becomes singular.

Another means of improving tracking over the standard manipulator Jacobian variation is to substitute the Generalized Jacobian Matrix for the standard Jacobian. Like the reaction compensation variation, this controller accounts for the effect of the moving base. In this variation, the momentum constraints are built into the GJM allowing the controller to “predict” the effect that the joint velocity commands will have on the base motion and compensate immediately. Generally, this variation should offer a quicker response to base motion effects than the reaction compensation method which uses current measurements of base motion. Using the GJM incurs a higher compu-

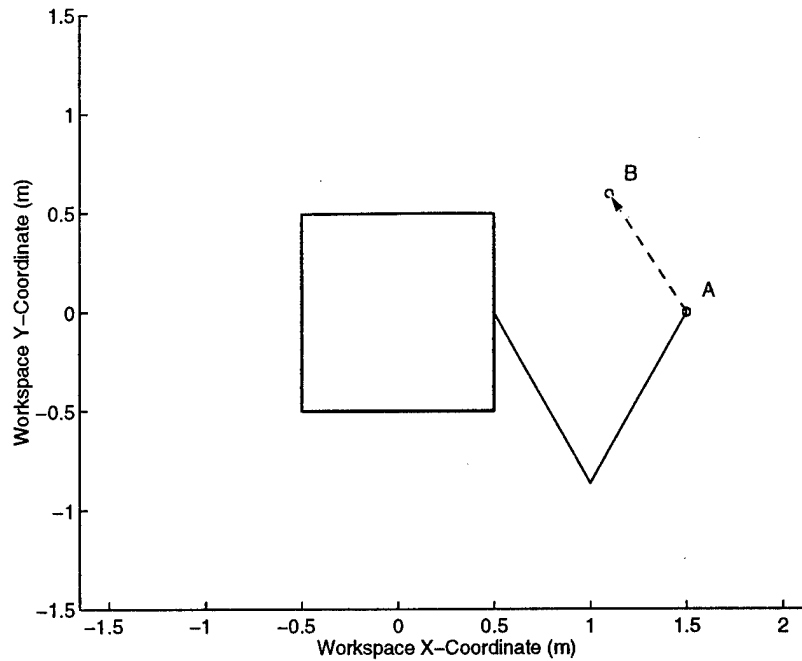


Figure 19. A Simple SMM Maneuver (Maneuver One)

tational cost than the first two methods, since the system mass matrix must be computed at each step in addition to the kinematic Jacobian. The accuracy of this method also depends on the fidelity with which the mass properties of each body in the system are known. This variation is also vulnerable to singularities. In this case, they are the dynamic singularities of the system rather than the kinematic singularities of the arm.

To demonstrate the performance differences between the controllers, consider the following example. Given the two-link (RR) planar SMM described in Chapter 4 (see Table 1), suppose the desired maneuver is to move the end-effector from point A to point B along a straight line as quickly as possible, starting with the system configured as shown in Figure 19. This is essentially a step input to the end-effector position.

The maneuver was simulated using each of the controllers in turn. To highlight the path-tracking differences in the controllers, the gains K_1 and K_2 were selected in each controller to give

Controller Variation	K_1	K_2	$\int \text{Torque } (N \cdot m \cdot s)$	$\int \text{Path Error } (m \cdot s)$
Generalized Jacobian	0.75	50	17.14	0.0602
Std. Jacobian w/Base Motion Fdbk	0.75	50	16.79	0.1147
Standard Jacobian	2.00	50	18.26	0.2574

Table 4. Controller Gains and Integral Metrics for Linear Trajectory (Maneuver One)

approximately equal settling times. The gains are shown in Table 4. The end-effector position time history is given in Figure 20, showing the settling time of all three controllers to be about seven seconds. Interestingly, the torque requirements for performing this maneuver are also essentially equal. Figure 21 shows the root-sum-squared value of the joint torques during the maneuver, and the integral of these curves is given in Table 4. For each controller variation, the torque is initially high and then quickly drops off, just as one would expect for a step input. Each controller variation is the highest and lowest of the three at some time during the maneuver, but the integral metric indicates that all variations expend similar total levels of energy to complete the maneuver.

The equal cost (torque integral) for equal performance (settling time) suggests that there is no important difference between the controllers. However, the third metric does reveal an important distinction between the controller variations. Figure 22 shows the end-effector path for each case. In this example, the end-effector is expected to move a straight line distance of about 0.7 meters, and the controller based only on the standard Jacobian swings up to 0.2 meters off the path at one point. The reaction compensation method fares better with a maximum path error of about 0.06 meters. The controller variation using the GJM provides the most accurate tracking of the desired path, with a maximum path error of less than 0.02 meters. The integral of the path error magnitude over the entire maneuver is given in Table 4, with this metric indicating that the GJM variation tracks twice as well as the reaction compensation variation and four times better than the naive variation. *The*

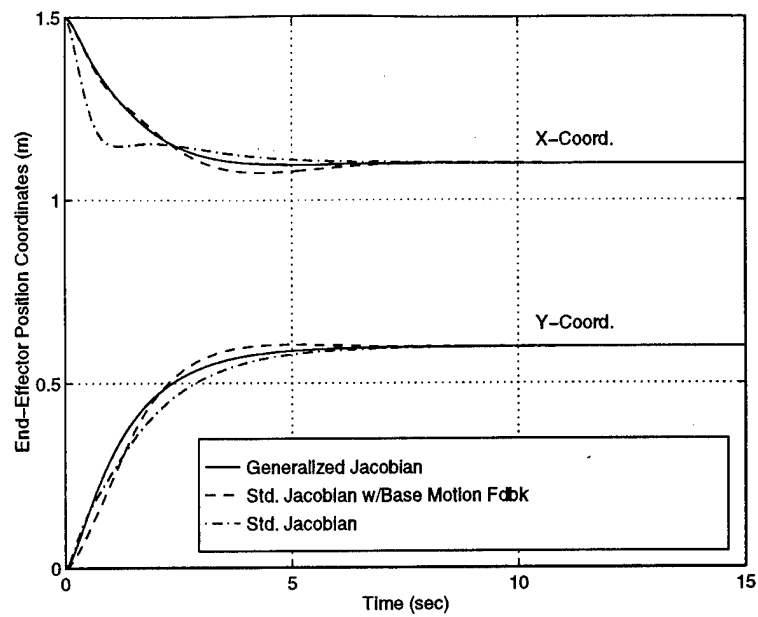


Figure 20. Response of Free-Floating SMM to an End-Effector Position Step Input

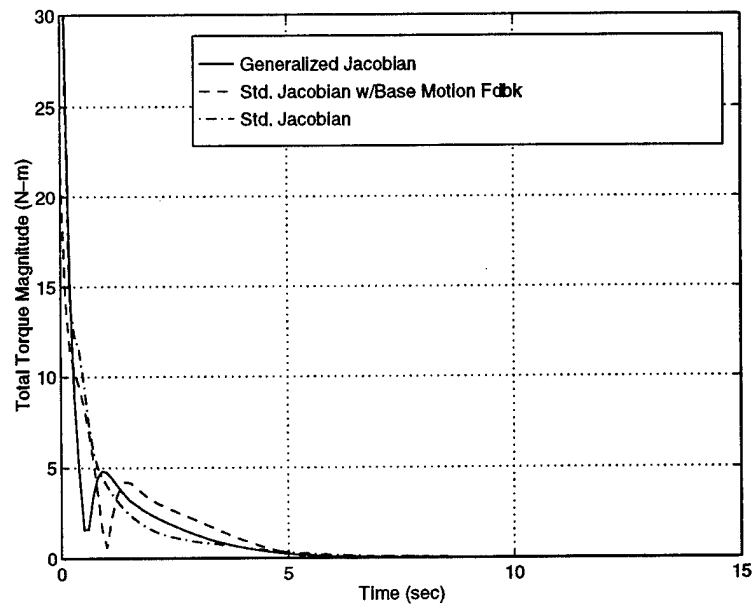


Figure 21. Torque History for Position Step Input

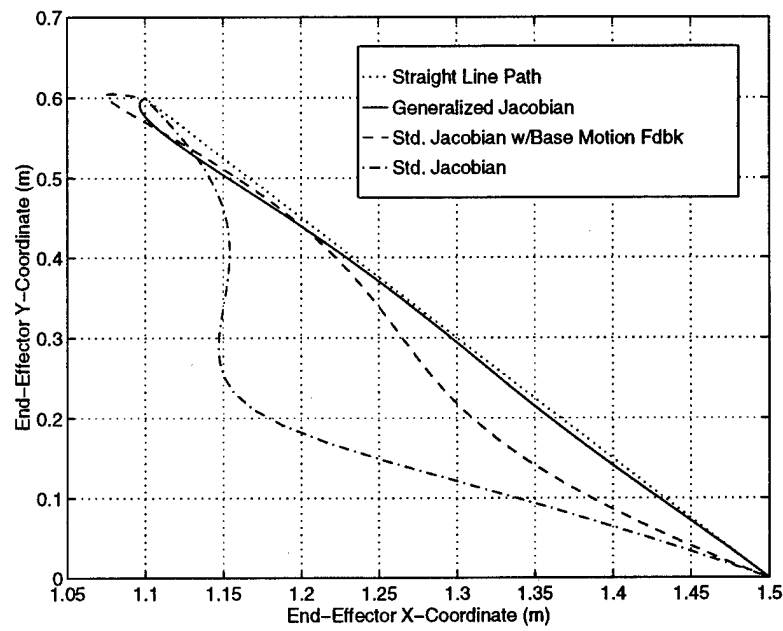


Figure 22. End-Effector Path for Position Step Input

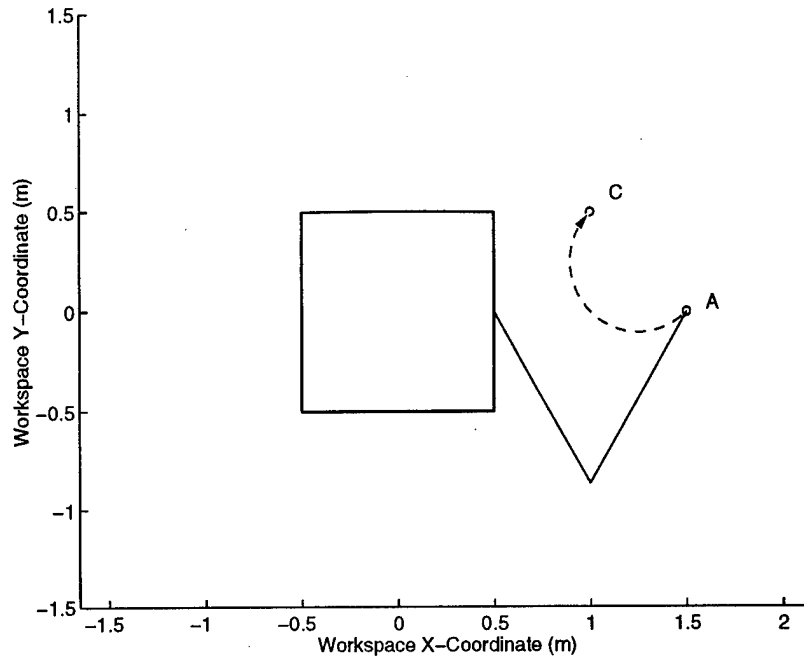


Figure 23. Circular Path Maneuver (Maneuver Two)

improved tracking at no extra cost is a strong incentive for using momentum-constrained Jacobians when controlling SMMs with unactuated degrees of freedom in the base.

We now consider a more complex maneuver. In this maneuver, the end-effector must follow a circular arc as shown in Figure 23. The path is to be traversed at a constant rate, arriving at the final point at time $t = 10$ seconds. This maneuver differs from the first both in the path type, circular as opposed to straight, but also in that it is position ramp input rather than a position step input. This is important because the initial accelerations are much lower in this maneuver than in the first.

As before, the controller gains (Table 5) were tuned to equalize response times as much as possible. This is shown in Figure 24, which plots the distance of the end-effector from the final position at all times during the maneuver. Each controller guides the end-effector to a point about 0.13 meters away from the desired final point at $t = 10$ seconds, and settles at about $t = 15$ seconds.

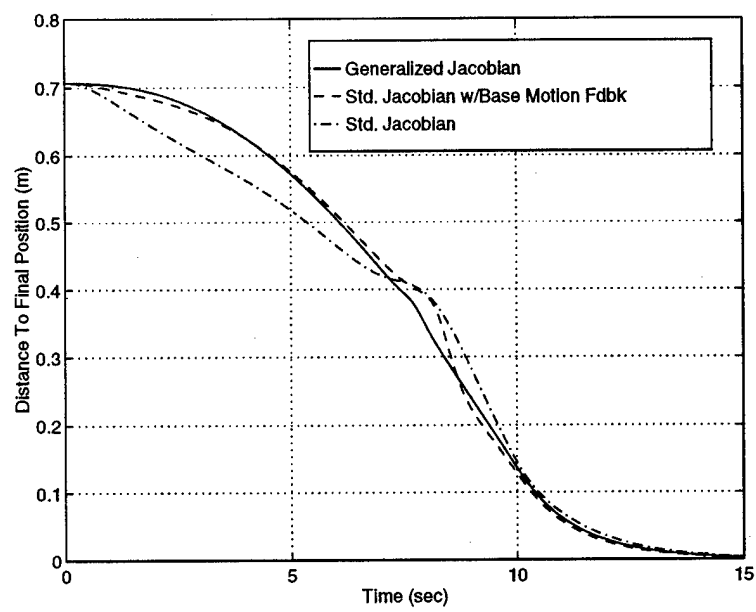


Figure 24. Time History of Distance to Final Position

Controller Variation	K_1	K_2	$\int \text{Torque } (N \cdot m \cdot s)$	$\int \text{Path Error } (m \cdot s)$
Generalized Jacobian	0.75	50	29.36	0.1537
Std. Jacobian w/Base Motion Fdbk	0.75	50	27.64	0.1577
Standard Jacobian	2.00	50	17.55	0.5282

Table 5. Controller Gains and Integral Metrics for Semi-Circular Trajectory (Maneuver Two)

The time history of the torque magnitude for each controller is shown in Figure 25. The most obvious feature of this figure is the high torque of the controllers in the middle of the maneuver. This is caused by a critical point in the maneuver where the arm is folded back on itself (joint two near 180°), resulting in very poor manipulability of the arm. This condition is exacerbated by the base velocity causing the end-effector to move in nearly the opposite direction to what is required at this point in the maneuver. The result is that the two controllers which recognize the base motion effect make very large torque demands at this point in the maneuver. The naive controller reaches a peak due to the low manipulability but does not recognize the base motion conflict compounding the problem, so it has a much lower peak torque. The integrals of the torque magnitude are given in Table 5.

The resulting end-effector paths are shown in Figure 26. The naive controller clearly gives very poor tracking throughout the maneuver, with significant deviations between 0.05 and 0.1 meters occurring at several points. The reaction compensation and GJM controller variations were better, both having consistent path errors of about 0.02 meters. The reaction compensation controller suffered a little more at the critical mid-point, deviating from the circular trajectory by about 0.04 meters at this point, but the path error integrals given in Table 5 show that the latter two controllers tracked equally well overall.

Examining the results of both example maneuvers, there is no clearly superior controller. In the first example, the GJM variation is obviously best, providing the best path following with both other

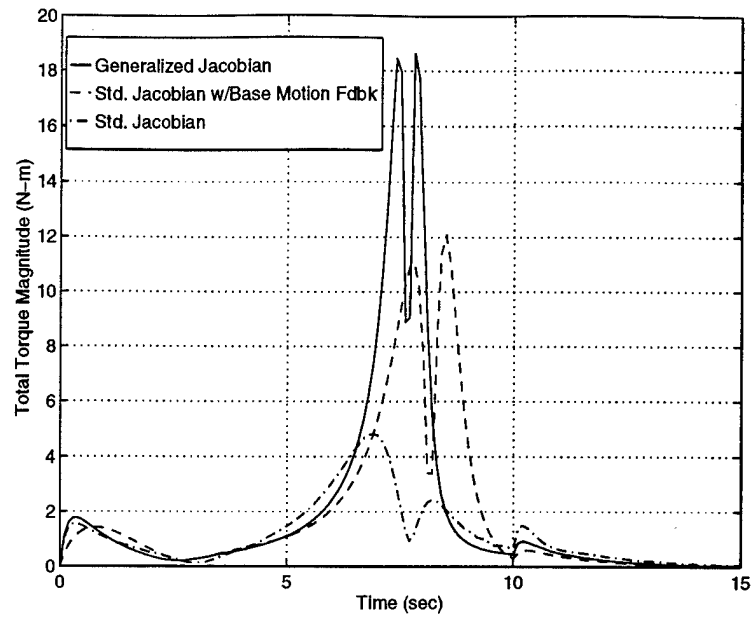


Figure 25. Torque Requirements for Circular Path (Maneuver Two)

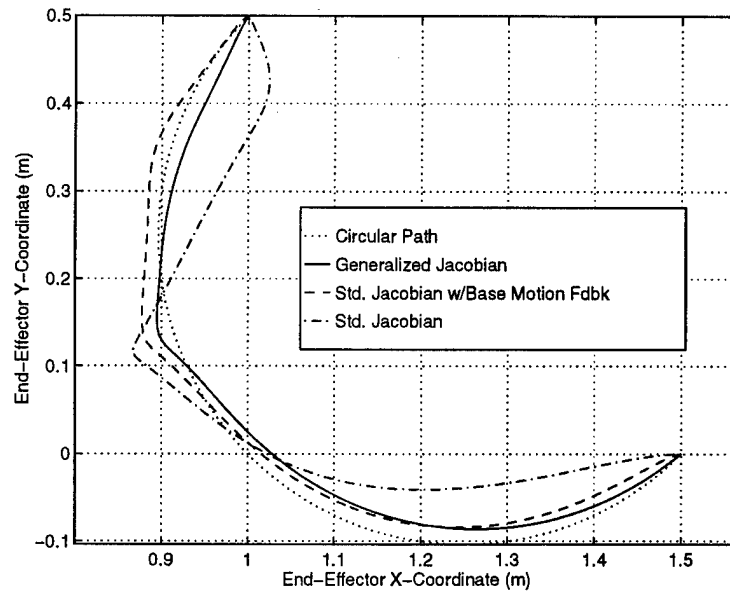


Figure 26. End-Effector Path For Circular Trajectory (Maneuver Two)

metrics equal. However, in the second example, the benefit of the momentum-constrained Jacobian is not as clear. Compared to the controller using only the standard Jacobian, the GJM variation provides much better path following, but at a significant cost in torque requirement. The preferred controller would depend on the relative weight of the metrics in the overall design. The GJM variation was actually slightly worse than the reaction compensation variation in the second example, requiring a little more torque for approximately equal tracking. The improvement of the reaction compensation method between the first and second examples can be attributed to the difference in the initial accelerations. In the first example, the outer loop controller demands a high acceleration of the end-effector. The reaction compensation controller reacts to the current base motion rather than predicting the effect of the joint velocity commands. When the base motion is changing rapidly, the controller performs poorly. In the second example, the initial end-effector acceleration requirement is much lower, resulting in lower base acceleration. The result is a significant improvement in tracking performance. Overall, the examples give only a slight edge to the GJM variation if all three metrics are of equal importance. However, if path following is a central concern over a variety of maneuver types, the GJM variation is the strongest choice. Since this will be true for many tasks, the GJM variation is considered the default free-floating SMM controller in subsequent sections of this work.

Each of the controller variations discussed above had the disadvantage of failing at singularities of the appropriate Jacobian. To demonstrate how this can affect even simple maneuvers, we consider a third example. Using the same SMM, the end-effector starts again at point A and must move along a straight line to point D, as shown in Figure 27. Using the basic SMM controller with the GJM variation results in the end-effector path shown as dashed line in Figure 28. Note that the end-effector never reaches the desired final position. During the maneuver, the system encounters a dynamic singularity causing the joint velocity commands to approach infinity. Controlled by the

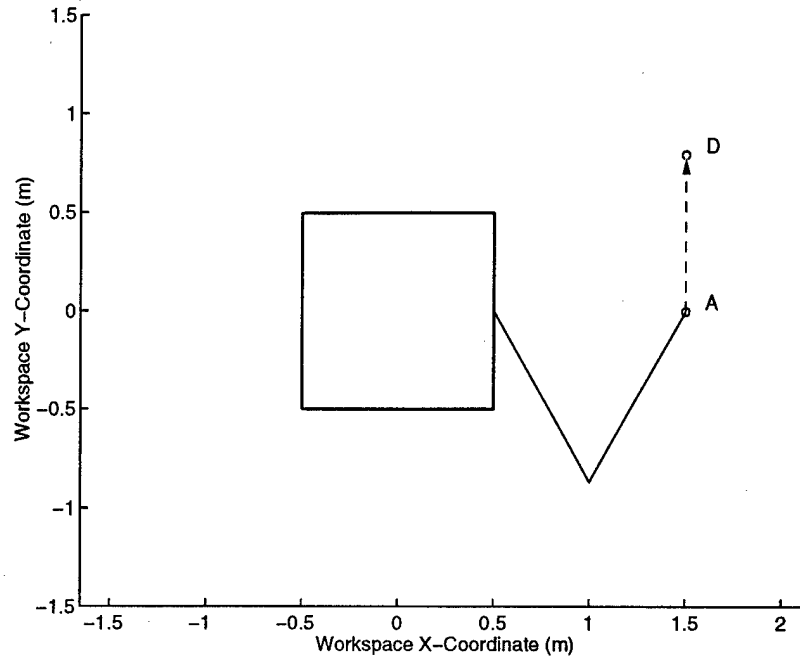


Figure 27. Simple Maneuver Prone to Singularity Encounters (Maneuver Three)

inner loop dynamics, the joints do not accelerate instantaneously to the commanded values, and as they ramp up, the primary component of end-effector motion is in a nonsingular direction. In the planar case, this means the end-effector accelerates rapidly in a direction nearly perpendicular to the singular direction. In an attempt to alleviate the effects of the dynamic singularity, Nakamura's Singularity-Robust Inverse (SR-inverse) [29] was used as an alternate means of inverting the GJM in the controller. The SR-inverse technique is similar to the pseudoinverse, but trades exactness of the solution for a more feasible solution in the neighborhood of singularities. The method is discussed in detail in Appendix D.1. Using the SR-inverse with the GJM produced the path represented by the solid line in Figure 28. The SR-inverse eliminates the excessive joint velocities, but the end-effector becomes trapped at the singularity and still does not reach the desired position.

The inability to reach the desired position in Maneuver #3 must be acknowledged as a failure of the basic SMM controller and all of its variations. This is a common failure mode of controllers

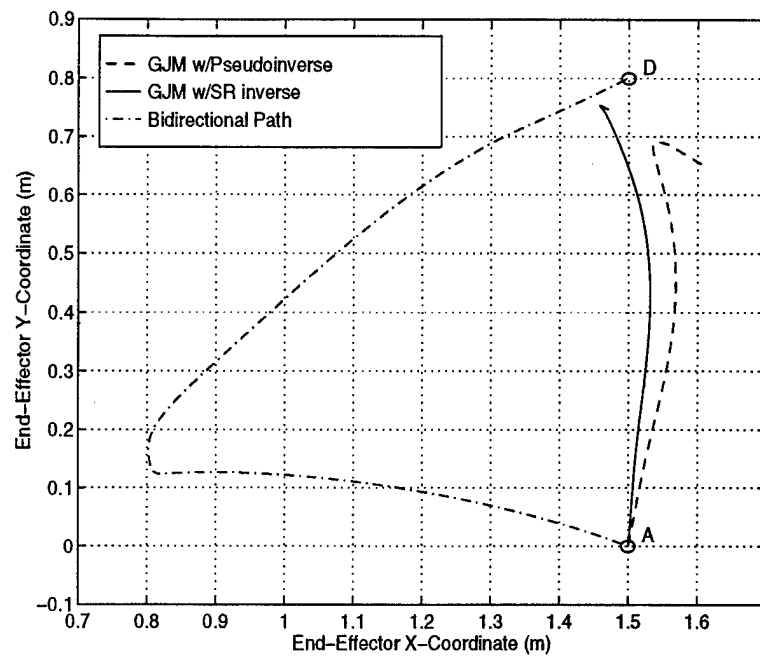


Figure 28. End-Effector Paths For Maneuver Three Using Free-Floating Control

which are based on inverting the manipulator Jacobian, and does not indicate that the target position is unreachable. Methods based completely in joint space are immune to this type of failure. By choosing an alternate trajectory in joint space, the end-effector can reach the desired point without encountering a singularity. The methods suggested by Reyhanoglu and McClamroch [39] or Nakamura and Mukherjee [30] offer the means of finding a possible trajectory. Unfortunately, both of these methods have two disadvantages. First, both methods are framed in joint space. As a result, solving the inverse kinematics problem to determine the joint states corresponding to the target end-effector position must be the first step in planning a suitable path. We noted earlier that finding solutions to the inverse kinematics is often quite complicated for fixed base robots, and adding the 6DOF base further clouds the issue. Occasionally, closed form solutions do not exist and the problem must be solved numerically. Even with a closed form solution, the inverse kinematics represents an additional computational load that must be addressed before starting the maneuver and precludes use in a teleoperated mode. Other computational requirements unique to each method are discussed below. The second major disadvantage of these methods is that neither method lends itself to precise tracking of a desired path between initial and final points in a maneuver. Controllers based on these methods will cause the end-effector to deviate significantly from a straight line path, making them less desirable in obstacle-dense environments or for tasks that require smooth, precise paths. The unique aspects of each method are addressed below.

Nakamura and Mukherjee's method is termed the "Bidirectional Approach" by the authors. The bidirectional approach synthesizes control inputs based on Lyapunov's direct method, using a quadratic form of the state (joint angles and attitude) errors as a Lyapunov function. A Jacobian similar to the GJM is used to relate the complete set of state velocities to the actuated state velocities. This Jacobian is constructed using the same momentum relations used in the GJM, but it does not include the kinematic relations between end-effector and system states. The algorithm assumes

two identical SMMs form a single dynamic system. One SMM starts at the initial state and one system starts at the final state. The error function used in the definition of the Lyapunov function is the difference between the states of the two “halves” of the system. Guided by Lyapunov’s direct method, the inputs are chosen to cause this error to converge to zero, forming a trajectory which starts at both ends of the desired maneuver and meets in the middle. This is the reason for the name of the method.

Details of our implementation of the Bidirectional Approach are given in Appendix D.2. Although Nakamura and Mukherjee argue that this method has less chance of encountering the null space of the Jacobian (which corresponds exactly to the dynamic singularities of the SMM), we found that the end states must be chosen with care to ensure the method converges to a solution. The most serious disadvantage of the method, however, stems from the metrics with which we judge a controller. Our requirement that the end-effector follow a particular path between points is very difficult to meet using the bidirectional approach. This is best seen by example. Maneuver #3 represents a worst case scenario, where we know the method must contend with a dynamic singularity. The dash-dotted line in Figure 28 shows the bidirectional solution to Maneuver #3. Clearly, this method does not satisfy the path following requirement.

The Reyhanoglu and McClamroch algorithm consists of four steps. These steps can be associated with paths in joint space, as shown for a two-link SMM in Figure 29. In step one, the robot is driven to the desired joint angles without regard for the base attitude. This implies that the end-effector position is uncontrolled at this stage in the algorithm. At step two, a closed path in joint space is computed that will exactly change the base attitude by the difference between the desired attitude and the attitude resulting from step one. The computational requirement for this step is not excessive, but it must be met before the system can proceed. The manipulator joints are then driven to the nearest corner of the closed path (a rectangular circuit is suggested by the authors) in

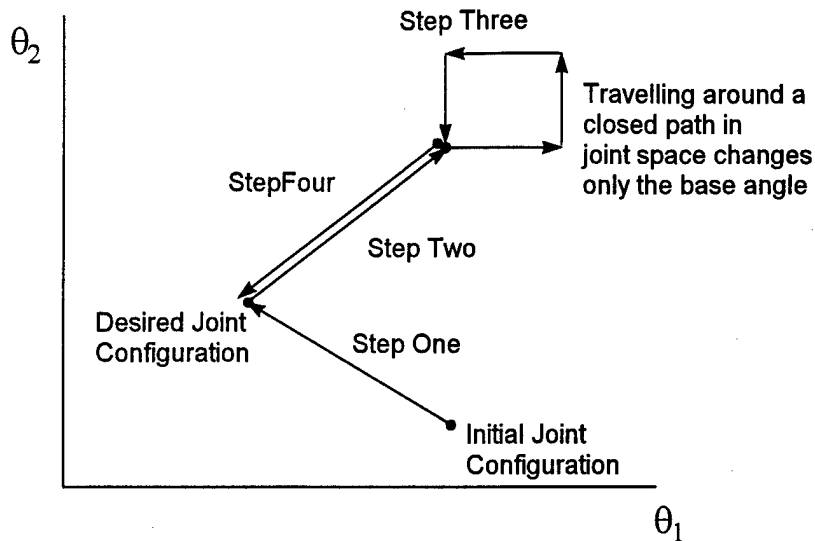


Figure 29. Example Path in Joint Space Using Reyhanoglu and McClamroch Control Scheme

joint space. Step three involves moving the joint angles through the previously computed closed path. Step four exactly reverses step two, leaving the system at the desired final state. Clearly, the end-effector path over the course of these four steps will be quite convoluted. Since this method obviously cannot precisely track a path between the initial and final points, its application to our example Maneuvers was not considered.

The significant advantage of this method is that it is completely immune to the effects of singularities. For this reason, it must be seriously considered if the path error metric is not a concern. This may be the case if the controller is only required to perform point-to-point tasks in an obstacle-free environment.

In order to find a suitable controller for performing tasks in all areas of workspace, including the Path Dependent Workspace, we must find another means of working around dynamic singularities. Since it was shown in Chapter 4 that dynamic singularities are eliminated by actuating the satellite base's attitude, we consider this approach next.

5.3 Base-Attitude Control (BAC) Case

5.3.1 Simple BAC Controller

The basic SMM controller can be used for an SMM with base attitude actuations with a few small modifications and assumptions. First, we assume that the actuation is provided in the form of external torques on the SMM. This implies the use of thrusters as the actuator type. The reason for this assumption at this point is that it allows us to focus on the advantages of base-attitude control without the concern of added dynamic effects that would result from using reaction wheels or control moment gyroscopes. Second, the variables $\dot{\theta}$ and $\dot{\theta}_c$ as used in the basic SMM controller block diagram (Figure 17) are now to be understood to include both the manipulator joint velocities and the base angular velocity. Finally, the Jacobian whose inverse is used to find $\dot{\theta}_c$ from \dot{r}_c is not the GJM, but rather a new Jacobian constrained only by the linear momentum equations. This is necessary since the system angular momentum is not conserved when the angular velocity is directly actuated. This form of Jacobian was developed in Section 4.6.2 (Eq. (142)). In the simple BAC controller, the inversion of the linear momentum constrained Jacobian is accomplished using the standard pseudoinverse solution.

Let us now reconsider Maneuver Three (Figure 27). Using a free-floating SMM, this maneuver proved to be quite difficult. The GJM controller variation failed completely due to the dynamic singularities, and although the bidirectional approach did allow the free-floater to reach the target point, it could not follow a straight line path. Using an SMM with the same inertial properties (Table 1 in Chapter 4), but with thrusters providing base attitude actuation, the maneuver was simulated again with the simple BAC controller. The resulting end-effector path is shown in Figure 30, along with the bidirectional path for reference. The SMM with base-attitude control has no singularity problems, and successfully tracks the straight line path to the target position. In addi-

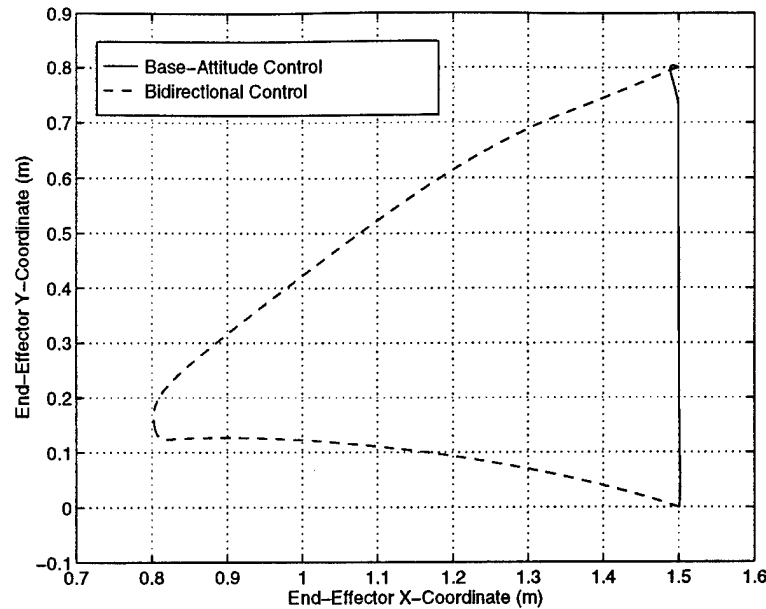


Figure 30. End-Effector Paths For Maneuver Three Using Simple Base-Attitude Control And Bidirectional Approach

tion, the base-attitude control incurs significantly lower torque costs¹¹ as shown in Figure 31. The torque magnitude of the base-attitude controlled SMM shown in the figure includes the base torque as well as manipulator joint torques. *The lower torque requirement combined with clearly superior tracking performance demonstrates the powerful advantage base-attitude control holds over all the free-floating methods.*

The most prevalent argument against base-attitude control is that the base torques may in fact be “costlier” than joint torques. Whereas joint torques are generated by electric motors which use a renewable energy source, creating large base torques with thrusters uses irreplaceable fuel. This translates to a much shorter life span or an enormous weight penalty in added fuel. A solution is to use the inherent redundancy when both base and arm are actuated to reduce the base torque requirements. In the example above, a standard pseudoinverse was used to translate end-effector velocity

¹¹Since the bidirectional approach is similar to a ramp input in that it does not require the large initial accelerations associated with a step response, the BAC controller was given a ramp input in generating the results shown in Figures 30 and 31. This allows a fairer comparison of torque requirements of the two methods.

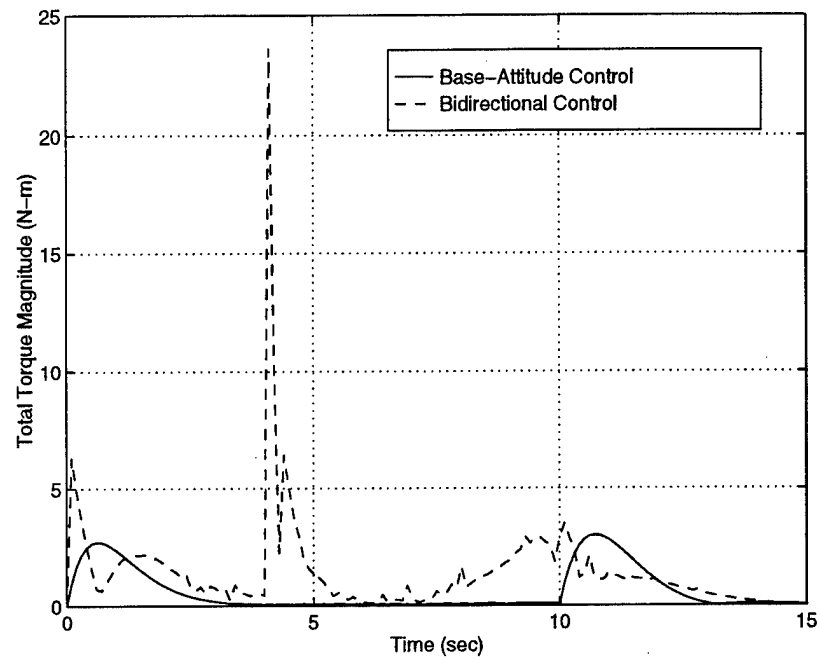


Figure 31. Total Torque Requirements For Maneuver Three Using Simple Base Attitude Control (Ramp Input) And Bidirectional Approach

commands into joint and base-attitude velocity commands. This can result in excessive use of the base control. A weighted pseudoinverse is an obvious remedy, but cannot actually provide much help. A weighted pseudoinverse rewards or penalizes base motion without necessarily reducing base torque requirements. To see this, consider the effects of first an infinite penalty on the base velocity and then second, an infinitesimally small penalty on base velocity. In the first case, the controller would attempt to fix the base, which would require significant base torque to counteract the internal torque caused by moving the manipulator joints. In the second case, the controller would attempt to move the base at high rates, which would also require significant base torque. A novel solution, which we term Reduced Base Torque Control, is based on using the angular momentum relations with Nakamura's task priority method [29]. The details of the RBTC method are developed in the following section.

5.3.2 Reduced Base Torque Control

The central concept of reduced base-torque control (RBTC) is to control the SMM so that it performs much like a free-floater until a dynamic singularity is approached. At this point, attitude control is phased in to continue moving the end-effector smoothly through the troublesome region. This concept is implemented by choosing the joint space velocity command, $\dot{\theta}_c$, using a multiple task priority method. The simple BAC solution is the first task, and is always given the highest priority. This ensures smooth motion of the end-effector, since this solution is unaffected by dynamic singularities as a result of the base control (recall the arguments in Chapter 4). The secondary task is to choose joint space velocities consistent with a zero angular momentum state for the system. For the rest-to-rest type of maneuvers expected for the SMM, system angular momentum will start and end at zero. Since angular momentum is conserved only if no external torque is applied, this task is equivalent to choosing system motion which does not require base control. If the second priority

task was completely met throughout a maneuver, the SMM would behave as if it were free-floating, and no base torque would be used. The commands resulting from the second task are in the null space of the first task, so they cannot adversely affect the end-effector path.

To develop this method, we first define the system inputs, \dot{q} , as the base angular velocity and the joint velocities,

$$\dot{q} = \begin{bmatrix} \omega \\ \dot{\theta} \end{bmatrix} \quad (231)$$

The first task is defined by the end-effector velocity, \dot{r} , which can be given in terms of the inputs by the relation

$$\dot{r} = J_1 \dot{q} \quad (232)$$

where the task one Jacobian, J_1 , is defined by

$$J_1 = \begin{bmatrix} J_\omega - J_v P_v^{-1} P_\omega & J_\theta - J_v P_v^{-1} P_\theta \end{bmatrix} \quad (233)$$

Recognize that J_1 is the linear momentum constrained Jacobian for the SMM from Eq. (142). The second task is to choose \dot{q} such that the system has zero angular momentum. The task two relation which corresponds to Eq. (232) for task one is

$$0 = J_2 \dot{q} \quad (234)$$

where J_2 is

$$J_2 = \begin{bmatrix} H_\omega - H_v P_v^{-1} P_\omega & H_\theta - H_v P_v^{-1} P_\theta \end{bmatrix} \quad (235)$$

The quantity $J_2 \dot{q}$ represents the total angular momentum (h) of the SMM from Eq. (109), also constrained by the linear momentum equation, Eq. (101) due to the lack of base translation actuation.

Now Nakamura's method of choosing \dot{q} according to task priority can be applied. Let \dot{q}_1 represent the inputs required to accomplish task one, and \dot{q}_2 represent the additional velocity added to achieve task two. Then

$$\dot{q} = \dot{q}_1 + \dot{q}_2$$

$$\begin{aligned}
&= J_1^\# \dot{r} - \hat{J}_2^\# (J_2 J_1^\# \dot{r}) \\
&= (J_1^\# - \hat{J}_2^\# J_2 J_1^\#) \dot{r}
\end{aligned} \tag{236}$$

where $\#$ denotes the pseudoinverse and $\hat{J}_2 = J_2(U - J_1^\# J_1)$. This equation ensures that \dot{q}_2 is in the null space of J_1 , so that no velocity added for the second task interferes with the completion of the first task. See Ref. [29] for the details of this method.

The first priority task Jacobian, J_1 , does not have dynamic singularities, so these configurations cannot affect the smooth motion of the end-effector. However, using Eq. (236) can still lead to problems at dynamically singular configurations. At a dynamic singularity, it is impossible to arbitrarily choose the system angular momentum using the null space of J_1 , or equivalently, \hat{J}_2 becomes singular. When the SMM approaches these configurations, accomplishing the second task without affecting the first task is only possible with very large inputs, so \dot{q}_2 approaches infinity. These inputs do not affect the first task, but nonetheless are impractical at the least and very undesirable. To remedy this problem, the input from the second task can be constrained in magnitude while keeping the same direction. In essence, the system prefers a state of zero angular momentum, but may not always achieve it. This allows external torques to drive the system while near a dynamic singularity. This concept leads to a new equation for the input velocities,

$$\dot{q} = \dot{q}_1 + k\dot{q}_2 \tag{237}$$

$$\dot{q} = (J_1^\# - k\hat{J}_2^\# J_2 J_1^\#) \dot{r} \tag{238}$$

where $0 \leq k \leq 1$. The system angular momentum resulting from this choice of \dot{q} is greater than Eq. (236) ($k = 1$ case), but is still less than not using the second task at all ($k = 0$ case). This can be proven by the following argument.

First, write the system angular momentum, h , in terms first of the input velocities, \dot{q} , and ultimately, of the desired end-effector motion, \dot{r} .

$$\begin{aligned}
h &= J_2 \dot{q} \\
&= J_2 \left(J_1^\# - k \hat{J}_2^\# J_2 J_1^\# \right) \dot{r} \\
&= \left(U - k J_2 \hat{J}_2^\# \right) J_2 J_1^\# \dot{r}
\end{aligned} \tag{239}$$

Recall that $\hat{J}_2 = J_2 \left(U - J_1^\# J_1 \right)$. The null space projection matrix, $\left(U - J_1^\# J_1 \right)$, is idempotent and symmetric. An identity from Nakamura states that if $A \in \mathbb{R}^{n \times n}$ is idempotent and symmetric, then for any $B \in \mathbb{R}^{m \times n}$,

$$(BA)^\# = A(BA)^\# \tag{240}$$

Letting $J_2 = B$ and $\left(U - J_1^\# J_1 \right) = A$, this identity provides the relation,

$$\hat{J}_2^\# = \left(J_2 \left(U - J_1^\# J_1 \right) \right)^\# = \left(U - J_1^\# J_1 \right) \left(J_2 \left(U - J_1^\# J_1 \right) \right)^\# \tag{241}$$

which can be premultiplied by J_2 to give

$$J_2 \hat{J}_2^\# = J_2 \left(U - J_1^\# J_1 \right) \left(J_2 \left(U - J_1^\# J_1 \right) \right)^\# \tag{242}$$

To simplify the notation, define a vector, x , and a matrix, A , by

$$x \triangleq J_2 J_1^\# \dot{r} \tag{243}$$

$$A \triangleq \left(J_2 \left(U - J_1^\# J_1 \right) \right)^\# \tag{244}$$

Then using Eq. (242), Eq. (239) becomes

$$h = \left(U - k A^\# A \right) x \tag{245}$$

This can be divided into two orthogonal components

$$h = \left(U - A^\# A \right) x + (1 - k) A^\# A x \tag{246}$$

The magnitude of h is related to the magnitude of these components by

$$\|h\|^2 = \left\| \left(U - A^\# A \right) x \right\|^2 + \left\| (1 - k) A^\# A x \right\|^2 \tag{247}$$

$$= \left\| \left(U - A^\# A \right) x \right\|^2 + (1 - k)^2 \left\| A^\# A x \right\|^2 \tag{248}$$

Controller Variation	Maneuver One		Maneuver Two		Maneuver Three	
	K_1	K_2	K_1	K_2	K_1	K_2
RBTC	0.60	100	1.0	200	0.5	200
Simple BAC	0.60	100	1.0	200	0.5	200
GJM (Free-Floater)	0.75	50	0.75	50	-	-

Table 6. Controller Gains For Maneuvers One, Two, and Three

Clearly, $k = 1$ minimizes $\|h\|$. This corresponds to fully completing task two. When $k = 0$, $\|h\|$ is maximized with respect to k , given that $0 \leq k \leq 1$. This corresponds to the pseudoinverse solution of the first task, with no regard for task two.

The scale factor k is chosen dynamically, to aggressively perform task two when the system is not near a singularity of \hat{J}_2 , and to avoid excessive values for \dot{q}_2 when near singularities of \hat{J}_2 . A smooth function for k with these properties is the ratio of the current minimum singular value of \hat{J}_2 to the maximum minimum singular value over all possible configurations. That is,

$$k = \frac{\sigma(\hat{J}_2(\theta))}{\max_{\theta}(\sigma(\hat{J}_2(\theta)))} \quad (249)$$

The denominator is a constant for a particular SMM, and can be calculated numerically in advance for use in the controller. The scale factor k will be near one when the system is near the most favorable configurations and be near zero when the system is near the singular configurations.

The performance of the RBTC concept was investigated by simulating Maneuvers One, Two and Three using the RBTC and simple BAC controllers. For Maneuvers One and Two, the results are also compared to the free-floating GJM based controller. The controller gains were chosen to equalize the response times in the neighborhood of ten seconds. These gains are shown in Table 6.

The end-effector paths for Maneuver One are shown in Figure 32. All three controllers performed adequately, but the RBTC and GJM variations had significantly lower tracking error than the BAC variation. The integrals of the path error, given in Table 7, indicate that overall, the RBTC

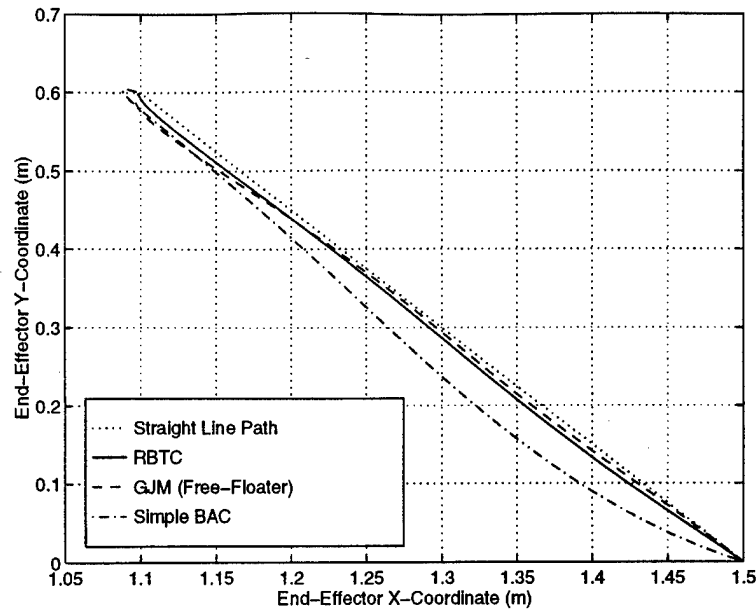


Figure 32. End-Effector Path For Position Step (Maneuver One)

and GJM controllers had nearly identical tracking. The tracking error of the BAC controller was approximately twice the error of the other two controllers. The larger error resulted from a direct conflict between the natural base reaction to the arm and the base angular velocity command generated by the BAC controller. All three controllers initially commanded positive joint velocities, which induced a negative base rotation in the free-floating SMM. The RBTC controller allowed this natural negative rotation to reduce base torque, while the simple BAC controller demanded a positive base rotation. Since the base was by far the largest body in the system, it takes longer to accelerate the base to its commanded velocity. This delay was the source of the tracking error.

The total torque magnitude for each controller is shown in Figure 33. This magnitude is the root-sum-squared value for the two arm joints and the base for the RBTC and BAC controllers, and for just the two arm joints for the GJM controller. The torque integrals are given in Table 7. For this maneuver, the RBTC performed the best, but this level of advantage should not be expected in every

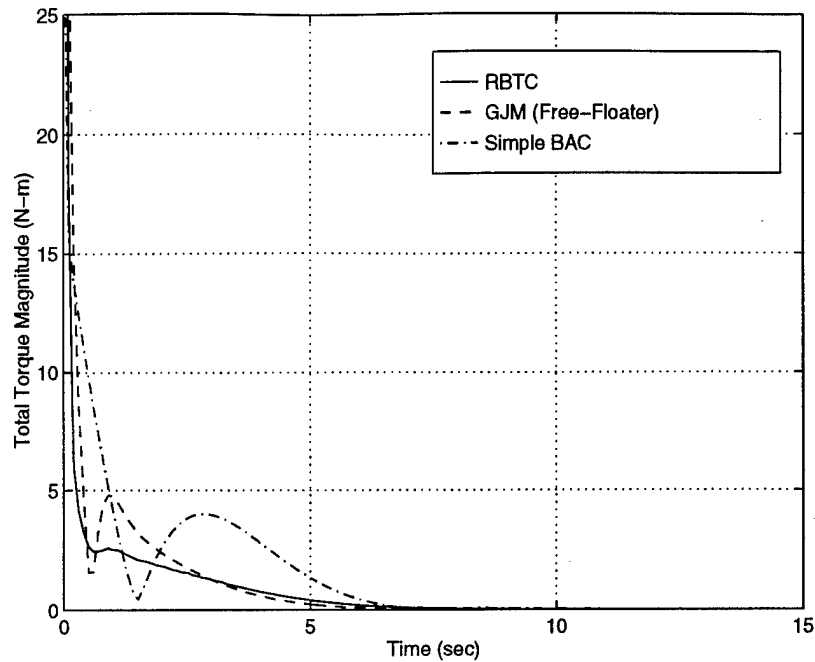


Figure 33. Total Torque Requirements For Position Step (Maneuver One)

case. The differences in total torque depend on the particular maneuver, starting configuration, and overall level of manipulability during the task. Our experience suggests that when the robot stays in regions of high manipulability for an entire maneuver, the free-floating method can produce lower torque results. More frequently, maneuvers take the system into areas of poor manipulability, closer to dynamic singularities. The resulting high velocities require higher accelerations and ultimately make the free-floater more expensive than the simple base controlled approach in terms of total torque. The RBTC is best able to move between the two approaches and so used the least total torque over a variety of maneuvers, although the actual degree of improvement varied widely.

By design, the RBTC controller was expected to yield an improvement in terms of base torque requirements. The base torque integral metric shows that the RBTC required less than a third of the base torque required by the simple BAC controller. Figure 34 shows the base torque history for each of the controllers. The initial negative spike in the RBTC base torque history is caused by the

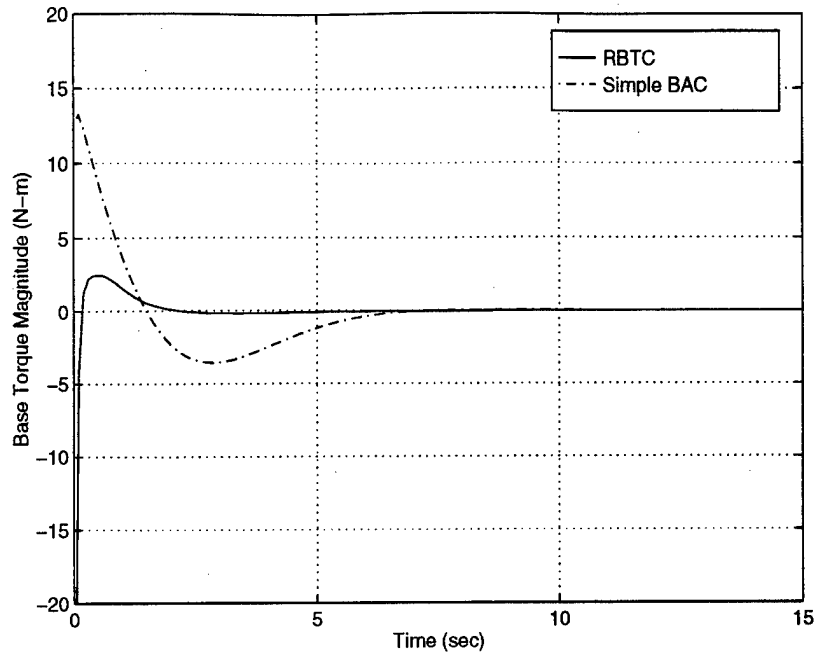


Figure 34. Base Torque Requirements For Position Step (Maneuver One)

lag in response between the inner and outer loops of the controller. The spike accelerated the base to the velocity consistent with the given joint commands and a zero (low) angular momentum state. Since a delay exists before the inner loop actually drives the joint to the commanded velocity, there is also a delay before momentum conservation drives the base to its commanded velocity. During this delay the inner loop recognizes the base angular velocity error and generates a momentary base torque. This combined effort to drive the base to its “natural” velocity explains why the spike is so short-lived. A more sophisticated version of the RBTC algorithm (perhaps inhibiting base torque for the first few controller cycles at the beginning of a maneuver) could eliminate this spike without significantly affecting the performance.

The results for Maneuver Two are given in Figures 35 through 37 and Table 8. For this maneuver, the tracking and total torque metrics are more even across the three controllers. The main performance difference between maneuvers One and Two is that due to the manipulability prob-

Controller Variation	\int Total Torque ($N \cdot m \cdot s$)	\int Base Torque ($N \cdot m \cdot s$)	\int Path Error ($m \cdot s$)
RBTC	15.30	5.475	0.0513
Simple BAC	22.98	19.56	0.1194
GJM (Free-Floating)	17.14	0.0	0.0602

Table 7. Integral Metrics for Maneuver One

lems of the free-floater in Maneuver Two, the simple BAC controller performs better than the GJM controller. Also note that with the lower initial accelerations required by this position ramp type maneuver, the tracking of the simple BAC controller is very comparable to the RBTC controller. As before, the base torque is significantly lowered by using the RBTC controller.

For Maneuver Three, only the RBTC and BAC controllers are shown, since it was seen earlier (Figure 28) that the free-floating controller options could not seriously compete in cases where dynamic singularities are present. Figures 38-40 and Table 9 show results similar to the results of Maneuvers One and Two. Total torque was comparable, with only a slight edge to the RBTC controller. The tracking error was significantly lower for the RBTC controller, as expected for a position step input like Maneuver One. The base torque requirement for the RBTC was less than half of the requirement for the simple BAC controller.

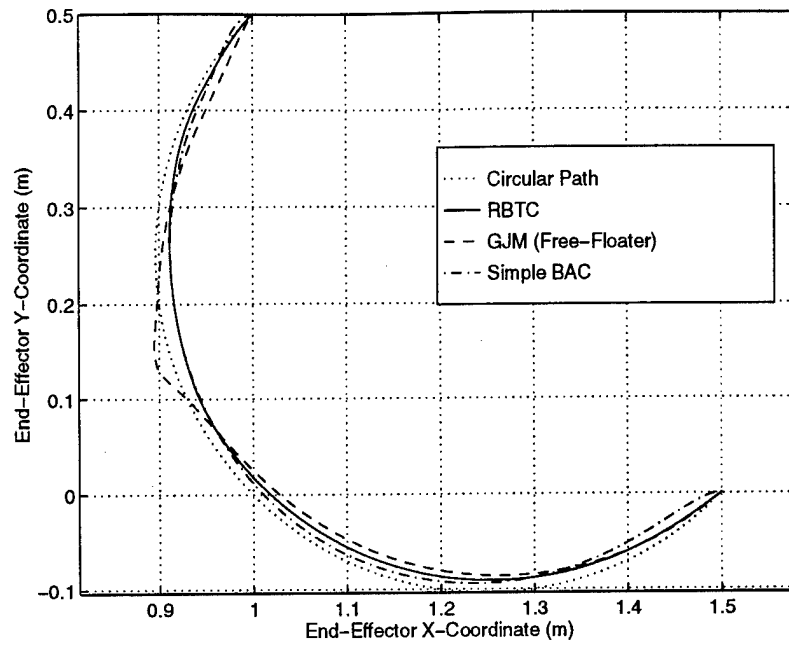


Figure 35. End-Effector Path For Circular Trajectory (Maneuver Two)

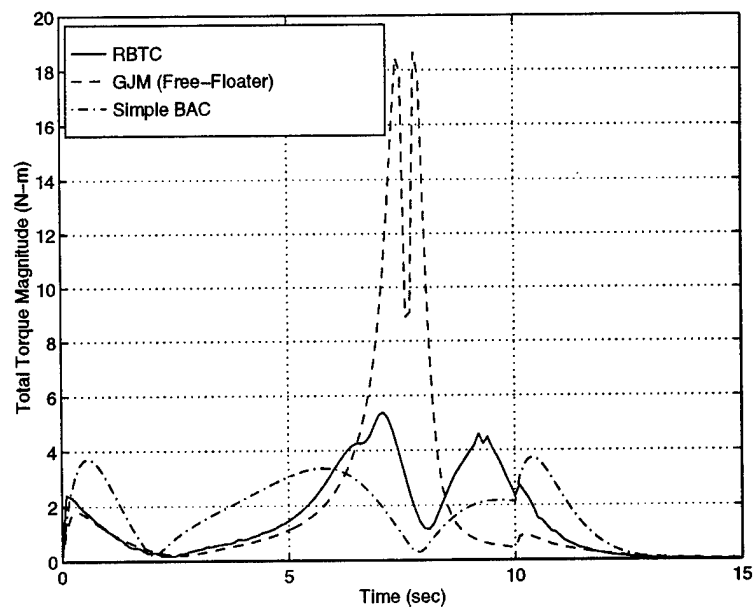


Figure 36. Total Torque Requirements For Circular Trajectory (Maneuver Two)

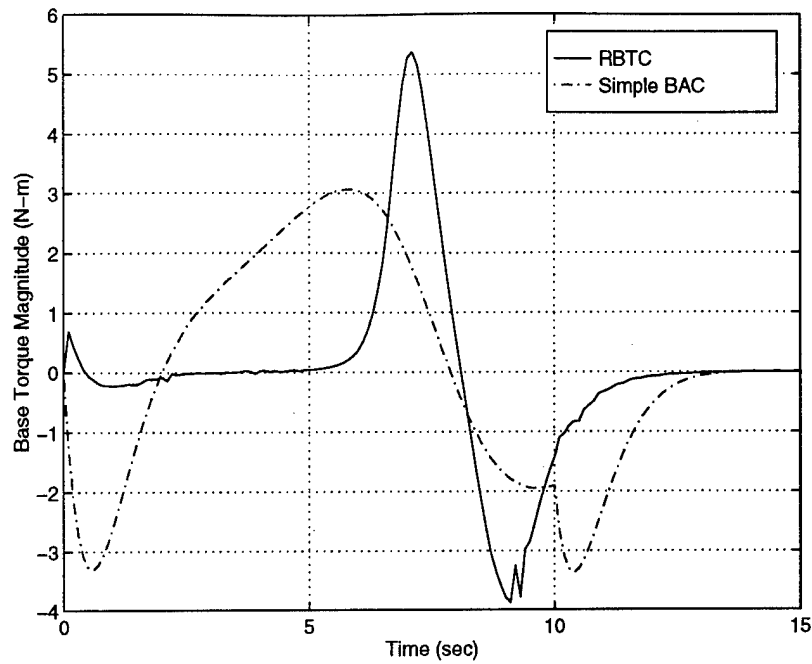


Figure 37. Base Torque Requirements For Circular Trajectory (Maneuver Two)

Controller Variation	\int Total Torque ($N \cdot m \cdot s$)	\int Base Torque ($N \cdot m \cdot s$)	\int Path Error ($m \cdot s$)
RBTC	22.72	12.06	0.1126
Simple BAC	25.14	22.57	0.1254
GJM (Free-Floating)	29.36	0.0	0.1537

Table 8. Integral Metrics for Maneuver Two

Controller Variation	\int Total Torque ($N \cdot m \cdot s$)	\int Base Torque ($N \cdot m \cdot s$)	\int Path Error ($m \cdot s$)
RBTC	29.30	12.07	0.0759
Simple BAC	32.56	28.30	0.1433

Table 9. Integral Metrics for Maneuver Three

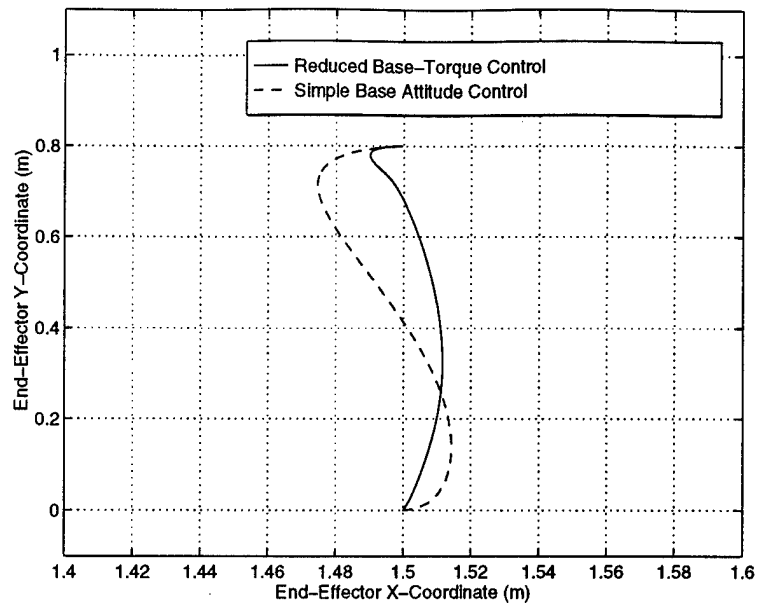


Figure 38. End-Effector Path For Singularity Prone Position Step (Maneuver Three)

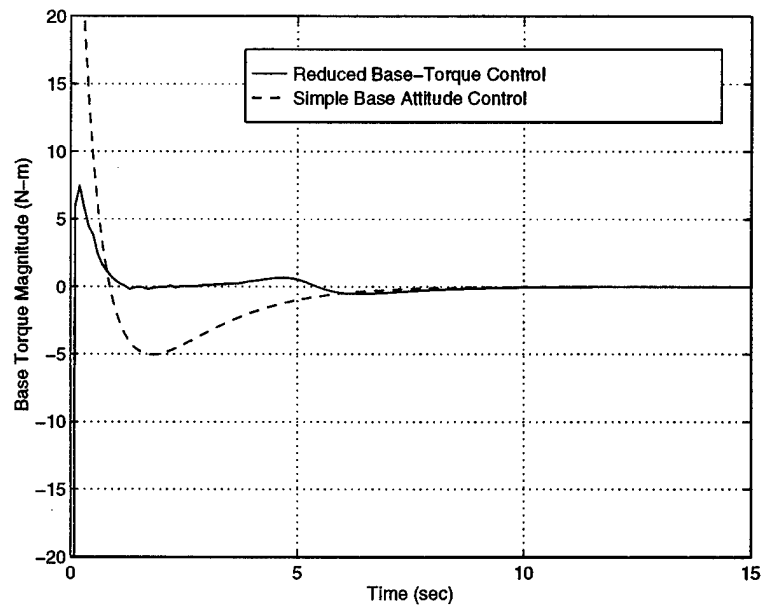


Figure 39. Base Torque Requirements For Maneuver Three

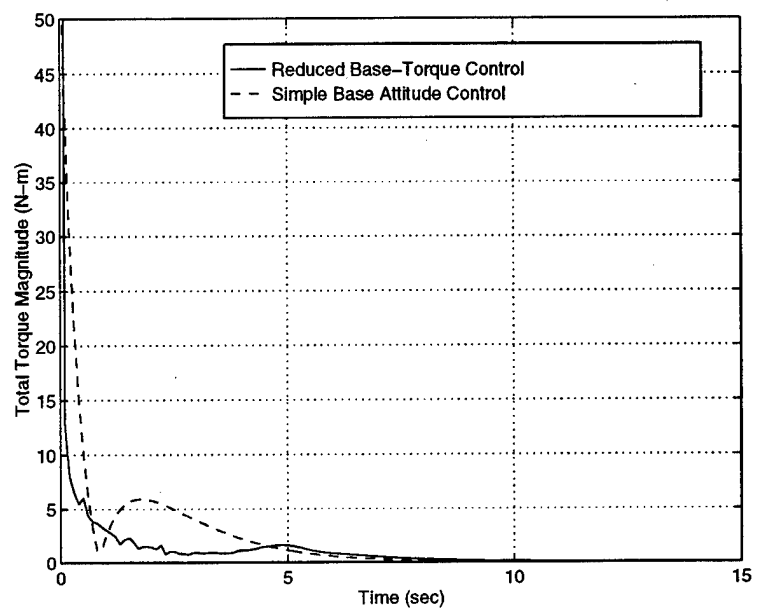


Figure 40. Total Torque Requirements For Maneuver Three

5.4 Summary

In Chapter 5, we extended the comparison of free-floating and base attitude control concepts by examining their performance in simple maneuvers, using a newly developed SMM controller. The new controller was designed to allow the incorporation of each of the control concepts without major modifications. The simulation results support several important conclusions:

1. Momentum constrained Jacobians provide better tracking than fixed base manipulator Jacobians, even when base motion feedback is used.
2. Free-floating control concepts using the GJM can fail in the neighborhood of dynamic singularities, even when using Nakamura's Singularity-Robust inverse.
3. Free-floating control concepts which rely on joint space trajectory planning methods are not well suited to tasks where precise path following is required.
4. The Base-attitude control concept performs well throughout reachable workspace in tracking and time response, but can result large base torques. The method is unaffected by dynamic singularities.
5. Base torque cost, in terms of the integral metric can be significantly lowered using the new Reduced Base-Torque Controller (RBTC).

Chapter 6 - Base Attitude Control Using Control Moment Gyros

The primary goal of this work is to demonstrate the importance of base control in an SMM system. Spacecraft attitude control can be accomplished by many means, both passive and active. However, to implement the envisioned base control concept, a powerful active attitude controller is required. There are two classes of actuators powerful enough to be considered viable options: thrusters and momentum exchange devices. Momentum exchange devices hold two distinct advantages over thrusters. First, they are more capable of providing smooth and precise input. In the base attitude control concept, spacecraft attitude is equivalent to the first joint of the manipulator. As such, it must be closely controlled to ensure proper end-effector motion. Second, momentum exchange devices use renewable electrical power instead of consuming fuel. Over the lifetime of an SMM, this advantage could result in very significant mass savings.

Momentum exchange devices are generally divided into two types: reaction wheels (RW) and control moment gyros (CMG). CMGs are preferable to reaction wheels because of their large torque capability, which allows lower weight, power, and size requirements for the same performance [4]. Among CMGs, one can choose between double-gimbaled control moment gyros and single-gimbaled control moment gyros. Single-gimbaled CMGs are commonly chosen over double gimbaled CMGs because of their relative mechanical simplicity and their advantage of transmitting the output torque through gimbal bearings rather than through a gimbal motor [24,36]. A disadvantage of single-gimbaled gyros is that for any cluster, there exist gimbal configurations at which an instantaneous loss of torque capability in some direction occurs. These gimbal configurations are known as singularities of the cluster, and can cause significant difficulties in controllers that are not designed to handle them.

In this chapter, we incorporate single-gimbaled CMGs into our earlier dynamic models and controllers. We consider the problem of cluster singularities and demonstrate the interaction between the SMM controller and the singularities through a simulated three-dimensional system. We introduce a new controller which avoids both gyro singularities and SMM dynamic singularities. Similar to the RBTC controller of Chapter 5, this controller uses a secondary task based on angular momentum to move smoothly between the base-controlled and free-floating modes of operation.

6.1 Adding a CMG Cluster to the n -Link SMM Model

6.1.1 Dynamic Model

A typical single-gimbaled CMG consists of a rotor mounted on a gimbaled frame, as shown in Figure 41. The rotor is spun at a constant high rate about its symmetric axis, generating a large fixed magnitude angular momentum vector. The direction of this momentum is varied by rotating the gimbal. A cluster of CMGs is used to create a variable source of angular momentum in the body frame. By controlling the cluster momentum a mechanism for spacecraft attitude control is provided. Since the cluster is fixed to the spacecraft base, changes in the cluster momentum induce equal and opposite changes in the spacecraft momentum. These angular momentum changes affect the spacecraft motion in a manner similar to external torques, although significant nonlinear effects are also introduced.

Adding a cluster of m CMGs to the n -link SMM model of Chapter 3 significantly increases the complexity of the system model. Each CMG adds two DOF and one or two rigid bodies, depending on whether the mass of the gimbal frame is included. A straightforward application of Lagrange's equation would result in $2m$ additional 2^{nd} -order equations of motion, or $4m$ additional 1^{st} -order state equations.¹² However, in many analyses of CMG dynamics [3, 11, 16, 36], the cluster equations

¹²Many of these equations are unnecessary. Ford and Hall [10] provide a comprehensive treatment of the equations of motion using an Euler-Newton based approach, which results in $2m$ 1^{st} -order state equations for a cluster of m SGCMGs.

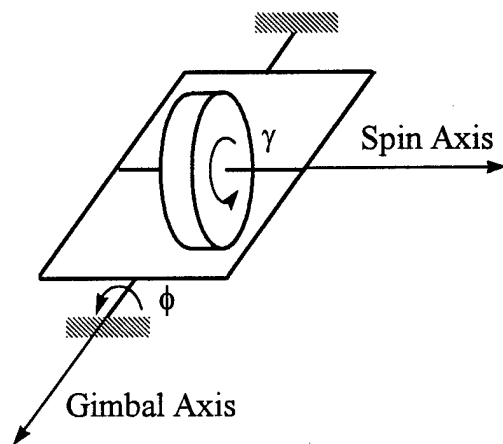


Figure 41. Single Gimbal Control Moment Gyro

of motion are simplified by retaining only the most dominant effects. Taking this approach, only m 1st-order state equations are necessary, and the SMM equations of motion derived in Chapter 3 remain largely unchanged. This simplification does not change the essential nature of the cluster singularities, and is adequate for our study of the interaction between CMG singularities and manipulator singularities.

Consider an SMM with a cluster of single-gimbal gyros. The total angular momentum of the system, h , may be written in the spacecraft body frame as

$$h = M_{\omega v}v_0 + M_{\omega}\omega_0 + M_{\omega\theta}\dot{\theta} + h_{\text{CMG}} \quad (250)$$

where v_0 , ω_0 , and $\dot{\theta}$ are the linear velocity of the base center of mass, the base angular velocity, and the arm joint velocities, respectively. The matrices $M_{\omega v}$, M_{ω} , $M_{\omega\theta}$ are submatrices of the system inertia matrix defined by Eqs. (60), (58), and (62) in Chapter 3, and h_{CMG} is the total angular momentum of the CMG cluster.

Differentiating Eq. (250) with respect to the inertial frame gives

$$\dot{h} = M_{\omega v}\dot{v}_0 + M_{\omega}\dot{\omega}_0 + M_{\omega\theta}\ddot{\theta} + \dot{M}_{\omega v}v_0 + \dot{M}_{\omega}\omega_0 + \dot{M}_{\omega\theta}\dot{\theta} + \dot{h}_{\text{CMG}} + \omega_0^\times h \quad (251)$$

When there are no external torques on the SMM, the total angular momentum of the system is constant, so $\dot{h} \equiv 0$. Then substituting Eq. (250) into Eq. (251) and rearranging terms gives

$$\begin{aligned} & M_{\omega v}\dot{v}_0 + M_{\omega}\dot{\omega}_0 + M_{\omega\theta}\ddot{\theta} \\ & + \left(\dot{M}_{\omega v} + \omega_0^\times M_{\omega v} \right) v_0 \\ & + \left(\dot{M}_{\omega} + \omega_0^\times M_{\omega} \right) \omega_0 \\ & + \left(\dot{M}_{\omega\theta} + \omega_0^\times M_{\omega\theta} \right) \dot{\theta} = -\dot{h}_{\text{CMG}} - \omega_0^\times h_{\text{CMG}} \end{aligned} \quad (252)$$

By defining C submatrices with the relation $C_x = \dot{M}_x + \omega_0^\times M_x$, this can be shortened to

$$M_{\omega v}\dot{v}_0 + M_{\omega}\dot{\omega}_0 + M_{\omega\theta}\ddot{\theta} + C_{\omega v}v_0 + C_{\omega}\omega_0 + C_{\omega\theta}\dot{\theta} = -\dot{h}_{\text{CMG}} - \omega_0^\times h_{\text{CMG}} \quad (253)$$

The lefthand side of Eq. (253) is identical to the angular velocity equations of motion derived earlier in Chapter 3. The right-hand side contains the terms related to the CMG cluster, forming an expression for the "effective torque" on the SMM base,

$$\tilde{\tau}_\omega = -\dot{h}_{\text{CMG}} - \omega_0^\times h_{\text{CMG}} \quad (254)$$

In Chapter 3, the complete equations of motion for the SMM were derived in the form

$$\begin{bmatrix} M_v & M_{v\omega} & M_{v\theta} \\ M_{\omega v} & M_\omega & M_{\omega\theta} \\ M_{\theta v} & M_{\theta\omega} & M_\theta \end{bmatrix} \begin{bmatrix} \dot{v}_0 \\ \dot{\omega}_0 \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} C_v & C_{v\omega} & C_{v\theta} \\ C_{\omega v} & C_\omega & C_{\omega\theta} \\ C_{\theta v} & C_{\theta\omega} & C_\theta \end{bmatrix} \begin{bmatrix} v_0 \\ \omega_0 \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \tau_v \\ \tau_\omega \\ \tau_\theta \end{bmatrix} \quad (255)$$

using a Lagrangian approach based on the kinetic energy of the system. Let us now consider how the kinetic energy is changed by introducing the gyro cluster. The kinetic energy is the sum of the kinetic energy in the base satellite, the manipulator links, and the CMG rotors (the gimbal mass is neglected).

$$T = T_{\text{SMM}} + T_{\text{rotors}} \quad (256)$$

The first term was developed in Section 3.3.5. The rotor term can be written

$$T_{\text{rotors}} = \sum_{j=1}^m T_j \quad (257)$$

where the energy of the j^{th} rotor is

$$T_j = \frac{1}{2} m_j \mathbf{v}_j \cdot \mathbf{v}_j + \frac{1}{2} \boldsymbol{\omega}_j \cdot \mathbf{I}_j \cdot \boldsymbol{\omega}_j \quad (258)$$

The velocity of the rotor center of mass, \mathbf{v}_j , is given by

$$\mathbf{v}_j = \mathbf{v}_0 + \boldsymbol{\omega}_0 \times \mathbf{p}_j \quad (259)$$

where \mathbf{p}_j is the position of the rotor center of mass relative to the base center of mass. This position is fixed in the \mathcal{F}_0 frame. The angular velocity of the rotor, $\boldsymbol{\omega}_j$, is given by

$$\boldsymbol{\omega}_j = \boldsymbol{\omega}_0 + \boldsymbol{\omega}_0^j \quad (260)$$

where $\boldsymbol{\omega}_0^j$ is the angular velocity of the rotor-fixed frame relative to the base fixed frame. The relative angular velocity, $\boldsymbol{\omega}_0^j$, includes the spinning of the rotor about the spin axis and the gimbal

rotational velocity. Substituting equations (259) and (260) into (258) and expanding, the kinetic energy of the j^{th} rotor becomes

$$T_j = \frac{1}{2}m_j [\mathbf{v}_0 \cdot \mathbf{v}_0 + 2\mathbf{v}_0 \cdot (\boldsymbol{\omega}_0 \times \mathbf{p}_j) + (\boldsymbol{\omega}_0 \times \mathbf{p}_j) \cdot (\boldsymbol{\omega}_0 \times \mathbf{p}_j)] \quad (261)$$

$$+ \frac{1}{2} [\boldsymbol{\omega}_0 \cdot \mathbf{I}_j \cdot \boldsymbol{\omega}_0 + 2\boldsymbol{\omega}_0 \cdot \mathbf{I}_j \cdot \boldsymbol{\omega}_0^j + \boldsymbol{\omega}_0^j \cdot \mathbf{I}_j \cdot \boldsymbol{\omega}_0^j]$$

Examining Eq. (261) reveals the necessary changes to the equations of motion. Since the rotor mass and position are constant in the body frame, the terms in the first set of brackets may be incorporated into the definition of the inertia submatrices M_v , $M_{v\omega}$, and M_ω , respectively. This is a simple matter of including the mass and center of mass position of the rotors when computing the mass and inertia of the spacecraft base. By extension, the submatrices C_v , $C_{v\omega}$, $C_{\omega v}$, and C_ω are also revised. The terms in the second set of brackets in Eq. (261) are associated with the terms \dot{h}_{CMG} and $\boldsymbol{\omega}_0^\times h_{CMG}$ in Eq. (253). The simplified model of the cluster momentum referred to earlier is equivalent to completely neglecting the energy term $\boldsymbol{\omega}_0 \cdot \mathbf{I}_j \cdot \boldsymbol{\omega}_0$ and neglecting the gimbal velocity contribution in $\boldsymbol{\omega}_0^j$. None of the terms in Eq. (261) involve arm joint positions or velocities, so the third row of Eq. (255) and all of the θ -submatrices are completely unchanged. The result is that the equations of motion for the SMM with a CMG cluster may be written as the $6 + n$ second order equations

$$\begin{bmatrix} M_v & M_{v\omega} & M_{v\theta} \\ M_{\omega v} & M_\omega & M_{\omega\theta} \\ M_{\theta v} & M_{\theta\omega} & M_\theta \end{bmatrix} \begin{bmatrix} \dot{v}_0 \\ \dot{\omega}_0 \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} C_v & C_{v\omega} & C_{v\theta} \\ C_{\omega v} & C_\omega & C_{\omega\theta} \\ C_{\theta v} & C_{\theta\omega} & C_\theta \end{bmatrix} \begin{bmatrix} v_0 \\ \omega_0 \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 \\ \tilde{\tau}_\omega \\ \tau_\theta \end{bmatrix} \quad (262)$$

Eq. (262) does not constitute a complete model of the system dynamics. The dynamics of the cluster angular momentum, h_{CMG} , is also required, since it is required to compute the effective torque, $\tilde{\tau}_\omega$.

In general, the dominant portion of the cluster angular momentum is the sum of the angular momenta of each rotor about its spin axis. The orientation of this axis in the body frame depends only on the associated gimbal angle, so that the cluster momentum can be written in the body frame

as

$$h_{\text{CMG}} = [h_1(\phi_1) + h_2(\phi_2) + \dots + h_m(\phi_m)] h_r \quad (263)$$

where h_r is the magnitude of the angular momentum of a single rotor about its spin axis, and the h_i are unit vectors in the direction of the spin axis. This expression neglects the relatively small portions of cluster momentum associated with the gimbal velocities, $\dot{\phi}$, and the spacecraft angular velocity, ω_0 . Some of the terms involving ω_0 are simply incorporated into the analysis by ensuring that the mass and mass center of each CMG is included in the mass and inertia matrix of the spacecraft base. Based on the expected operating range of gimbal rates and spacecraft angular velocities, the CMGs can be designed such that h_r is much larger than any of the neglected terms, so that the momentum computed using Eq. (263) does not differ significantly from the true momentum of the cluster.

The time derivative of the cluster momentum with respect to the spacecraft body frame is

$$\dot{h}_{\text{CMG}} = D(\phi) \dot{\phi} \quad (264)$$

where $D(\phi)$ is the Jacobian matrix

$$D(\phi) \triangleq \frac{\partial h_{\text{CMG}}}{\partial \phi} = \begin{bmatrix} \frac{\partial h_1}{\partial \phi_1} & \frac{\partial h_2}{\partial \phi_2} & \frac{\partial h_3}{\partial \phi_3} \end{bmatrix} \quad (265)$$

The problems of maintaining a constant rotor spin rate and quickly achieving desired gimbal rates can be addressed independently, allowing the gimbal rates, $\dot{\phi}$, to become the lowest level command input for the spacecraft attitude control. Then the m first-order equations required to complete the model of the system dynamics are the trivial equations

$$\dot{\phi} = \dot{\phi}_{\text{CMD}} \quad (266)$$

6.1.2 SMM Controller Using SGCMG Cluster for Base Actuation

In modifying the basic SMM controller of Chapter 5 for use with a CMG cluster, we begin by eliminating the base linear velocity equations from Eq. (262) as before (see Appendix A). By using

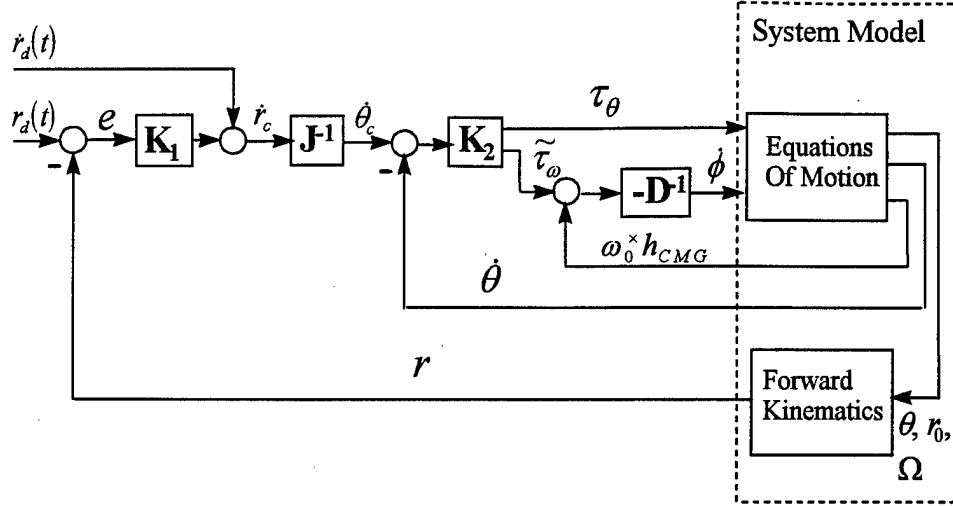


Figure 42. Controller for SMM with CMG Cluster

the linear momentum conservation relations, Eq. (262) can be reduced to

$$\tilde{M}\ddot{q} + \tilde{C}\dot{q} = \tilde{\tau} \quad (267)$$

where

$$\dot{q} = \begin{bmatrix} \omega_0 \\ \dot{\theta} \end{bmatrix} \quad (268)$$

and

$$\tilde{\tau} = \begin{bmatrix} \tilde{\tau}_\omega \\ \tau_\theta \end{bmatrix} \quad (269)$$

Eq. (267) is functionally equivalent to Eq. (189) in Chapter 5, so the basic SMM controller may still be applied. However, the control law Eq. (195) dictates $\tilde{\tau}$, and the control inputs for the SMM with CMG system are τ_θ and $\dot{\phi}$. The relation between the gimbal rates, $\dot{\phi}$, and the effective torque, $\tilde{\tau}_\omega$, must be incorporated into the controller. The gimbal rates necessary to impose $\tilde{\tau}_\omega$ are found by solving Eqs. (254) and (264) for $\dot{\phi}$,

$$\dot{\phi} = -D^{-1} (\tilde{\tau}_\omega + \omega_0^x h_{CMG}) \quad (270)$$

The result is the controller shown in the block diagram in Figure 42.

6.2 The Effect of CMG Cluster Singularities

Examining Eq. (270) it is clear that if $D(\phi)$ becomes singular, no solution for $\dot{\phi}$ will exist. Singularities of $D(\phi)$ are termed CMG cluster singularities, or simply CMG singularities. These are configurations of the gimbals in which there exists some direction for which no torque can be generated. These singularities are analogous to kinematic singularities associated with a manipulator Jacobian, where at certain joint configurations, no end-effector motion can be generated in some direction.

In order to examine the effect of CMG singularities on the operation of an SMM system, we consider a simple three-dimensional SMM, consisting of a three-link elbow manipulator mounted on a satellite base, as shown in Figure 43. Base attitude control is generated by a cluster of three SGCMGs, mounted inside the base in an orthogonal configuration, as shown in Figure 44. The simplified expression for cluster angular momentum, in which only the dominant terms have been retained, can be written in the base frame as

$$h_{\text{CMG}} = \begin{pmatrix} \cos(\phi_1) + \sin(\phi_3) \\ \sin(\phi_1) + \cos(\phi_2) \\ \cos(\phi_3) + \sin(\phi_2) \end{pmatrix} h_r \quad (271)$$

The physical parameters of the example SMM are given in Table 10.

The CMG Jacobian matrix associated with the example SMM is found by substituting Eq. (271) into Eq. (265), giving

$$D(\phi) = \begin{bmatrix} -\sin \phi_1 & 0 & \cos \phi_3 \\ \cos \phi_1 & -\sin \phi_2 & 0 \\ 0 & \cos \phi_2 & -\sin \phi_3 \end{bmatrix} \quad (272)$$

The simple geometry of this three CMG cluster enables one to easily find many singular configurations. One example of a singular CMG configuration for the given cluster geometry is $\phi_1 = \pi/2$, $\phi_2 = 0$, and $\phi_3 = \text{any value}$. In this configuration, there is no choice of gimbal velocities that can

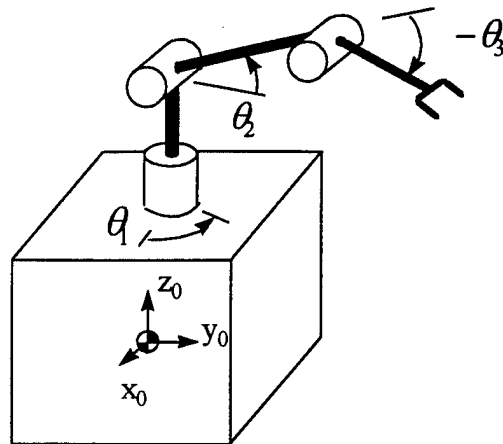


Figure 43. Satellite with Three-Link Elbow Manipulator

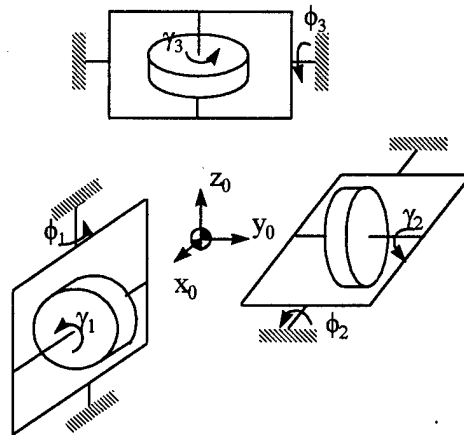


Figure 44. Orthogonally Mounted Three-CMG Cluster

Table 10. Physical Parameters for SMM with Three Link Elbow Manipulator

Body	I ($kg \cdot m^2$)	m (kg)	l (m)
0	$\text{diag}\{166.67, 166.67, 166.67\}$	1000	1
1	$\text{diag}\{0.0785, 0.3665, 0.3665\}$	15.71	0.5
2	$\text{diag}\{0.1571, 2.6965, 2.6965\}$	31.42	1
3	$\text{diag}\{0.1571, 2.6965, 2.6965\}$	31.42	1
Rotor Momentum: $h_r = 15.71 \text{ kg} \cdot \text{m}^2/\text{s}$			

move the cluster angular momentum vector in the y_0 -direction. The y_0 -component of the cluster angular momentum has reached the maximum possible magnitude. Many such singular configurations exist, forming a manifold in gimbal space. The common feature of these configurations is that they all represent the maximum possible magnitude of the cluster momentum in a particular direction. When the angular momentum of the cluster cannot be increased in some direction, no torque can be generated in this direction as long as the cluster configuration remains unchanged. Returning to the robot analogy, the singularity manifold is analogous to the workspace limit of a robot. Just as the manipulator Jacobian becomes singular at the workspace limit, the CMG cluster becomes singular when it reaches its torque production limit. Furthermore, in the same way that a redundant manipulator can encounter singularities inside its workspace, a CMG cluster containing more than three CMGs may encounter singularities before reaching a torque production limit. Indeed, in the CMG literature, singularities are divided into interior and outer singularities. Extensive research has been done in the area of CMG singularities, including important early work by Margulies and Auburn [24] and more recent work by Paradiso [36], Bedrossian *et al.* [3], and Ford [9]. These researchers studied methods of exploiting redundancy in the cluster to avoid interior singularities. Many of these methods are similar to those used for avoiding manipulator singularities in kinematically redundant robots. Our example system does not use redundancy in either the manipulator or the cluster, instead focussing on the redundancy which results from the interaction of the two.

The discussion above suggests that for a nonredundant CMG cluster, the singularities may be regarded as the “saturation” of the spacecraft attitude actuator. This insight motivated our investigation of the relationship between the occurrence of CMG singularities and the level of effort required from the cluster during a typical maneuver. Consider how the CMG configuration changes in response to a position step command. At the beginning of the maneuver, we assume the CMG cluster is in a zero momentum state. At first, the position error is large, causing the controller’s outer loop

to generate large command velocities for both the arm joints and the spacecraft angular velocity. The high angular velocity command dictates a large effective torque command, and ultimately, high CMG gimbal rates. The angular momentum of the CMG cluster begins to increase, approaching a maximum which will coincide with a singular condition. At some point in the maneuver, the position error decreases, leading to lower angular velocity commands and subsequently reducing the demands on the CMG cluster. Eventually, the cluster returns to a zero momentum state, far from a singular configuration.

True singularity problems arise only if the CMG cluster reaches a singular configuration (saturation point), causing the controller to request infinite gimbal velocities. In a step response, the initial peak demand on the CMG cluster will depend on the step size and the desired response time. If the position step command is too large or the outer loop gain, K_1 , is set too high, the CMG cluster will encounter a singularity.

For example, consider the SMM of Table 10 with the controller from Figure 42 given a position step command in which the end-effector begins at $r_i = (1.414, 0.0, 1.0)^T$ and is commanded to move to $r_f = (1.0, 0.0, 1.5)^T$. The relation between the outer loop gain and the CMG cluster singularity measure (the determinant of $D(\phi)$) is shown in Figure 45. If the outer loop gain was set to a value slightly higher than $K_1 = 0.12$, the CMG cluster would encounter a singularity when commanded to perform this simple position change. The corresponding end-effector position response is shown in Figure 46. Varying the step size using a constant value for K_1 results in a similar effect as seen in Figure 47. Step sizes marginally greater than 0.81 meters cause the CMG cluster to reach a singularity before completing the maneuver. The corresponding end-effector response is shown in Figure 48.

In both cases, high gains or large steps, the controller initially demands high angular acceleration of the spacecraft base. This requires a large initial torque, and when the CMG cluster is too

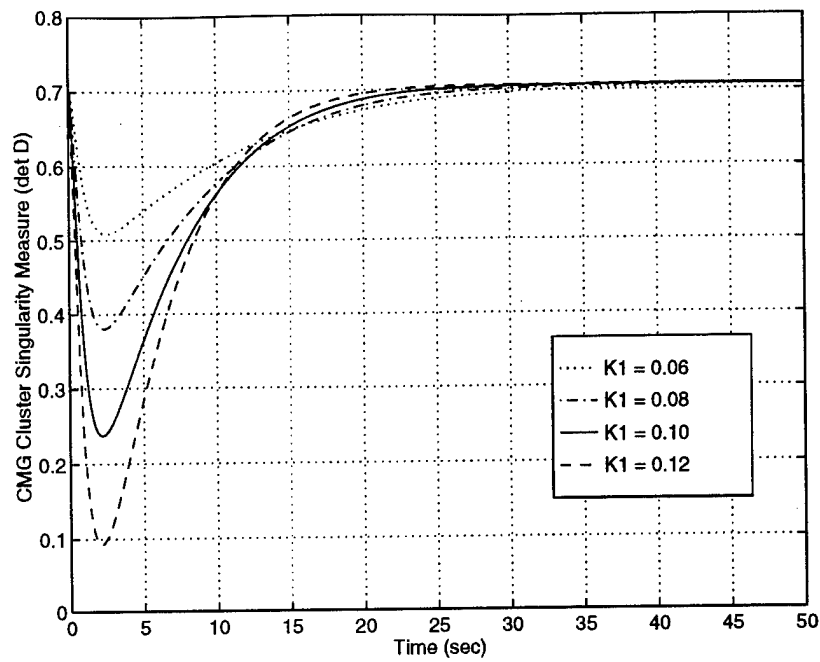


Figure 45. CMG Cluster Singularity Measure For Varied Outer-Loop Gain

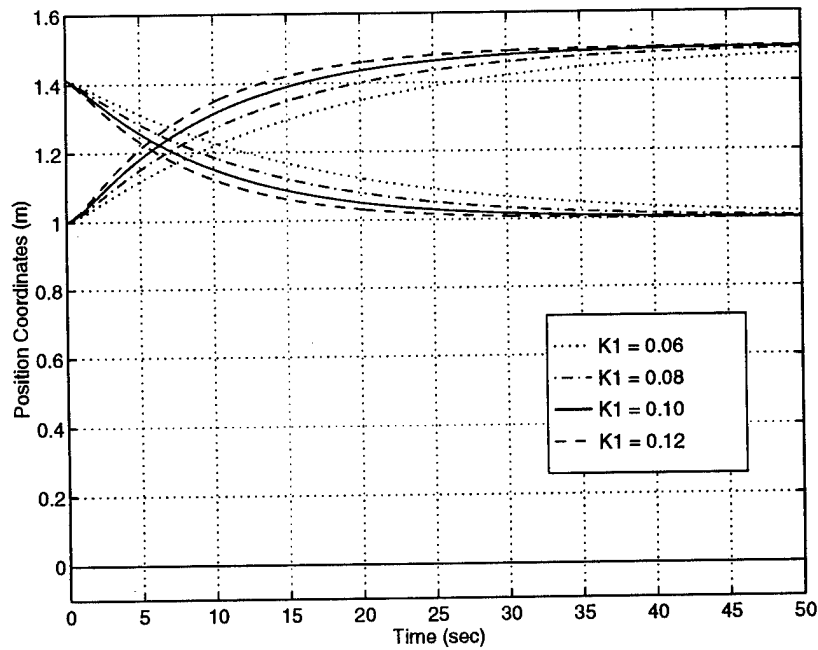


Figure 46. End-Effector Response For Varied Outer-Loop Gain

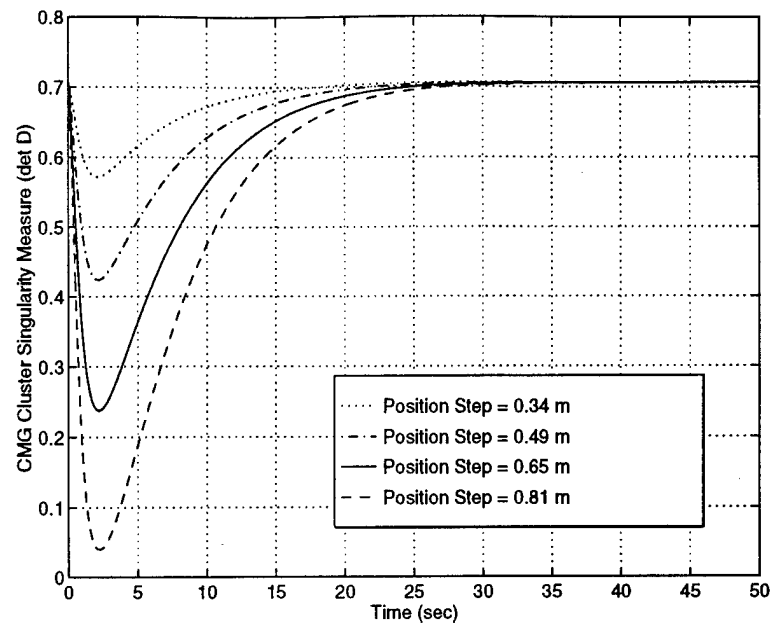


Figure 47. CMG Cluster Singularity Measure For Varied Position Step Size

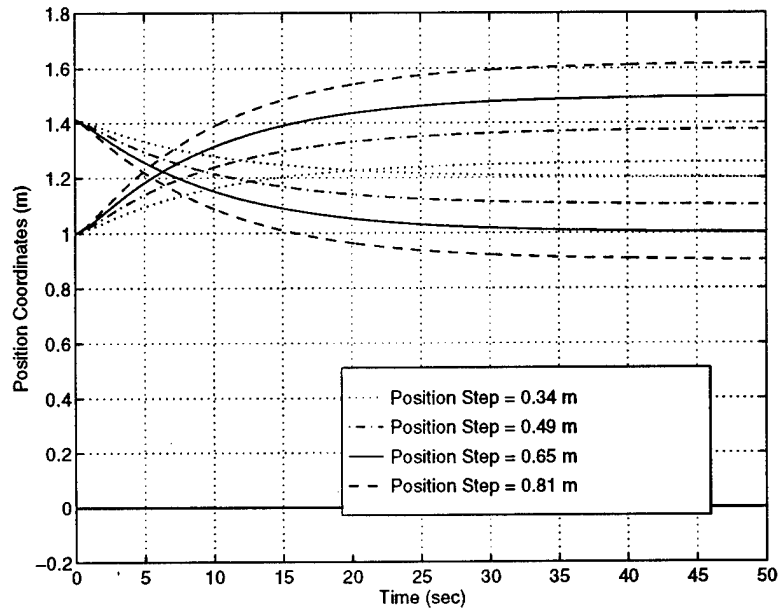


Figure 48. End-Effector Response For Varied Position Step Size

small to provide the peak torque required, it will saturate and become singular. This relation between speed of response and CMG singularities has been noted previously in studies of spacecraft attitude control by Hoelscher and Vadali [11] and Bedrossian *et al.* [3].

6.3 Controlling For Singularity Avoidance

6.3.1 Avoiding CMG Singularities

From these examples, it is apparent that in order to avoid CMG (outer) singularities, one must decrease the peak load on the CMGs. One way is to reduce the K_1 gains, but this will also increase the overall response time to any step input. A second option is to use the system redundancy provided by base control. Recall that in Chapter 5, we introduced the Reduced Base-Torque Controller (RBTC) and showed that it could decrease the base torque required for a maneuver. The fundamental idea of the RBTC was to mimic a free-floating SMM unless a dynamic singularity was encountered, at which point the base actuation would be phased in to eliminate the dynamic singularities. Under this philosophy, the base actuation is used only when necessary. While this appeared promising for the CMG singularity problem, our simulation experience shows that the RBTC is not a reliable means of avoiding CMG singularities. The reason is that although the RBTC tends to lower base torque for maneuvers in an integral sense, it does not generally lower peak base torque requirements. In fact, the peak torque requirement is frequently higher than that demanded by the basic controller. If thrusters are the primary base attitude actuators, the chief concern is saving fuel. Fuel usage relates directly to the integral of the torque, so the high peak torque is of less importance. However, when using CMGs for base actuation, lowering the total impulse may be beneficial in the sense that less total energy is consumed, but the most significant concern is avoiding the singular configurations, since these can cause a failure of the controller. Encounters with CMG singu-

ties are related to the peak torque requirements, so RBTC is not an appropriate choice of controller when using CMGs.

Although the RBTC is not suitable for CMG singularity avoidance, the redundancy arising from base control can still be exploited. However, instead of using base actuation sparingly, the CMG cluster must be considered an equal partner with the manipulator joints in driving the end-effector. Rather than phasing in base actuation to counter dynamic singularities, base actuation is only phased out when necessary to counter CMG singularities. This constitutes a fundamental shift in the way system redundancy is used, but is easily justified in the new context. The electric motors which power CMG gimbals are essentially equivalent to the arm joint motors, whereas thrusters are quite different in nature.

This strategy can be implemented by choosing appropriate joint space command velocities, \dot{q}_c , in the outer loop of the SMM controller. The controller generates a command for the end-effector velocity using the Resolved Motion Rate Control approach. The end-effector velocity is related to the joint space velocities through a Jacobian,

$$\dot{r} = J\dot{q} \quad (273)$$

where

$$\dot{q} = \begin{bmatrix} \omega_0 \\ \dot{\theta} \end{bmatrix} \quad (274)$$

for a base attitude controlled SMM. Choosing \dot{q} to achieve the desired \dot{r} requires some form of inversion of Eq. (273). In the block diagram shown in Figure 42, the inversion is represented by the " J^{-1} " block. However, a simple matrix inverse is not actually used in most of the controllers we have considered. In developing the RBTC in Chapter 5, \dot{q} was formed by summing two "tasks"

$$\dot{q} = \dot{q}_1 + k\dot{q}_2 \quad (275)$$

where

$$\dot{q}_1 = J_1^\# \dot{r} \quad (276)$$

represents the minimum norm (pseudoinverse) solution to Eq. (273) and

$$\dot{q}_2 = \hat{J}_2^\# J_2 J_1^\# \dot{r} \quad (277)$$

represents the additional joint space velocity needed to produce the free-floating solution. The definitions of J_1 , J_2 , and \hat{J}_2 are given in Section 5.3.2. The difference between the RBTC and the controller required for effective use of the CMG cluster lies in the method of choosing the scale factor k in Eq. (275). Since the CMGs are to be used equally with the arm joints whenever the cluster is not near a singularity, k should have a nominal value of zero. As the cluster approaches a singularity, k should increase. This will produce a \dot{q} which is closer to the solution for a free-floating SMM, and effectively reduce the need for the CMGs. Since \dot{q} is still an exact solution of Eq. (273), this does not affect the end-effector velocity, \dot{r} . Therefore, this method does not penalize the system response time as would a method based on decreasing K_1 .

An algorithm for k was heuristically determined, motivated by the thoughts above. First, a basic measure of CMG cluster singularity is computed as

$$d = |\det(D(\phi))| \quad (278)$$

The maximum value of d is one, which occurs when the spin axes are mutually orthogonal, and the minimum value is zero. This measure is then used in the formula

$$k = 1 - \left(\frac{d}{d + \varepsilon} \right)^{a/(d + \varepsilon)} \quad (279)$$

where the constants a and ε were chosen to shape the $k(d)$ function suitably. This algorithm provides a smooth relation between the CMG cluster singularity and the scale factor k , ensuring it is near zero over a wide range of CMG gimbal configurations but quickly rises when the cluster approaches a singularity. Figure 49 shows the $k(d)$ curve for the values $a = 50$, $\varepsilon = 0.0001$, which provided good results in our experience. The controller formed by taking the SMM (with CMGs) controller from

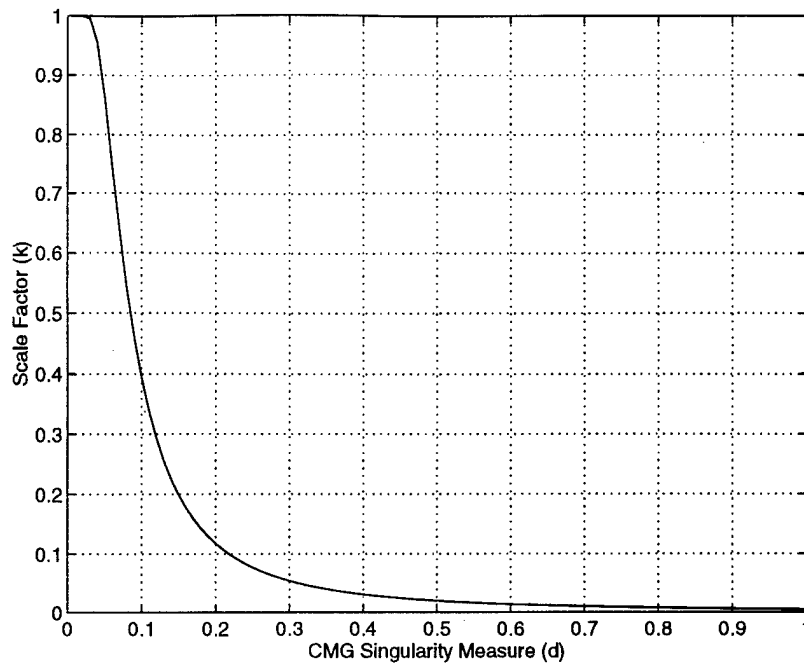


Figure 49. Free-Floating Task Scale Factor vs. CMG Cluster Singularity

Figure 42 and using Eqs. (273)-(279) to convert end-effector velocity commands to joint commands (the “ J^{-1} ” block) is termed the CMG Singularity Avoidance (CSA) controller.

Consider the simple point-to-point maneuver used in the earlier example. From the trend seen in Figure 46, it would appear that increasing the outer-loop gain to $K_1 = 0.14$ can be expected to improve the response time compared to the response with $K_1 = 0.12$. However, using the basic controller with this gain, the system encounters a CMG singularity before completing the maneuver. This could be reasonably predicted by examining Figure 45. Using the CSA controller, the outer-loop gain can be increased dramatically to perform the maneuver much faster without encountering a CMG singularity. Figures 50 and 51 compare the responses of the basic controller with $K_1 = 0.12$ and the new controller using $K_1 = 0.24$. The new controller gives a greatly improved response time even though the minimum value of the singularity measure is higher than that of the basic controller. This is a result of the system’s increased use of the arm joints, as shown in Figure 52.

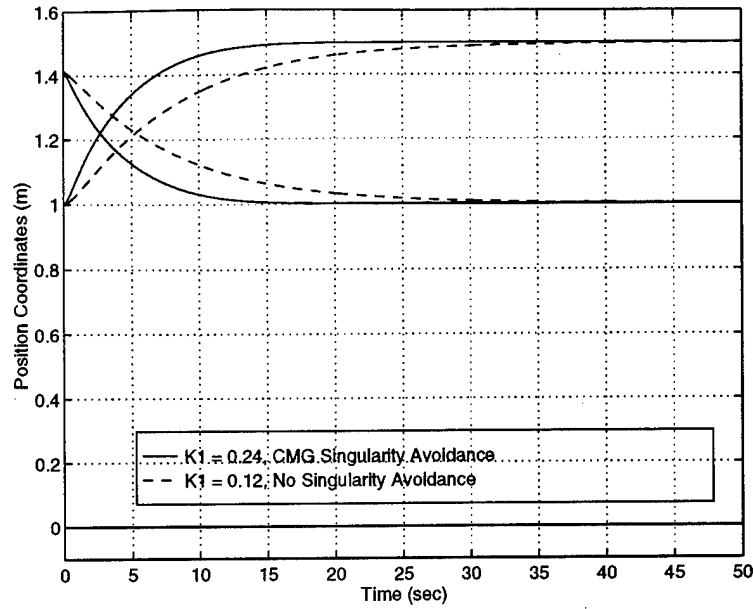


Figure 50. End-Effector Response For Controllers With And Without CMG Singularity Avoidance Term

When the system approached the CMG singularity, the CSA controller responded by making the arm joints take on a larger share of the load. The commanded joint space velocities not only ensure that the end-effector velocity is maintained, but also induces the CMGs to move back towards a zero-momentum configuration and away from the singular configuration. This can be seen in Figures 50 and 51, where the end-effector velocity (position slope) is still high at $t \approx 2$ seconds, while the CMG singularity measure has already begun to increase. If the scale factor k increased to one, the CSA controller would command joint space velocities corresponding to an SMM zero momentum state and the CMG cluster would be forced into its zero momentum state as well. This does not actually occur, because as the CMG moves away from the singular configuration, the scale factor k decreases according to Eq. (279).

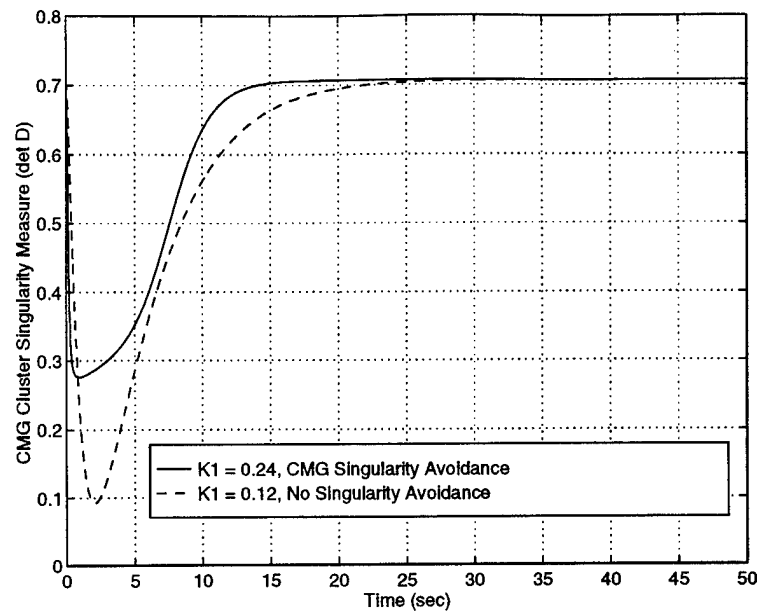


Figure 51. CMG Cluster Singularity Measure For Controllers With And Without CMG Singularity Avoidance Term

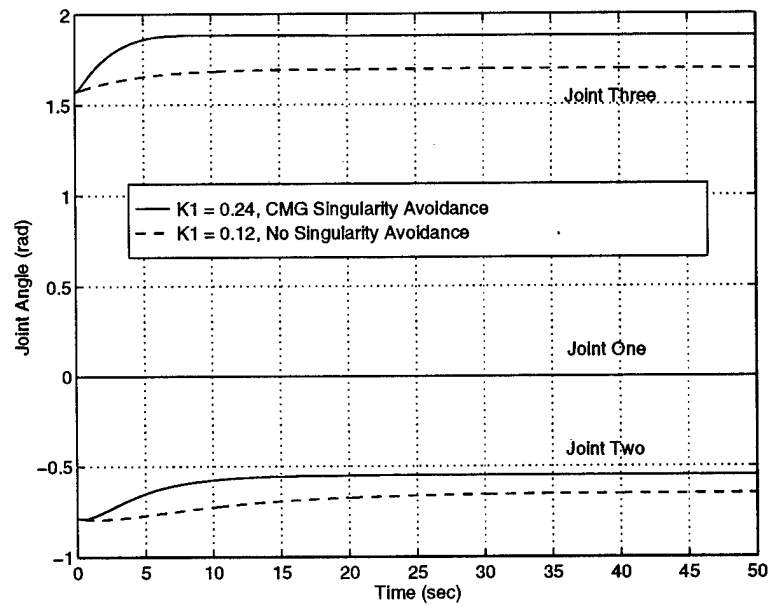


Figure 52. Joint Motion Using Controllers With And Without CMG Singularity Avoidance Term

6.3.2 Avoiding Combined Singularity Conditions (Dynamic and CMG Cluster)

The example above demonstrates the singularity avoidance properties of the CSA controller. However, we know from Chapter 4 that a free-floating mode of operation introduces the possibility of encountering dynamic singularities. Since the CSA controller seeks to emulate the free-floating mode when near CMG singularities, it is imperative that we consider the possibility that the system may encounter both types of singularities simultaneously. If this were to occur, the second term in Eq. (275) would approach infinity, leading to excessively high joint velocity commands. The trajectory chosen in the example lies entirely within the Path Independent Workspace of the SMM. Therefore, there was no danger of approaching a dynamic singularity while using the free-floating mode to avoid CMG singularities. For a more rigorous test, we consider another trajectory which will force the system into configurations where dynamic singularities can occur. The new trajectory begins at the same point as the first, $r_i = (1.414, 0.0, 1.0)^T$, but ends at $r_f = (1.5, 0.0, -1.0)^T$.

In order to judge how close the system is to a dynamic singularity, a metric must be chosen. A dynamic singularity was defined as a configuration of an SMM where the GJM becomes singular. Physically, these are configurations at which there exists no joint velocity set that can create an end-effector velocity in the singular direction. This occurs when the effects of the joint motion and the corresponding momentum conserving base motion exactly cancel. At these configurations, the secondary task Jacobian, \hat{J}_2 , also becomes singular. Since this matrix is computed for use in Eq. (275), it is convenient to base our dynamic singularity measure on it. The matrix \hat{J}_2 is not square, so we cannot use its determinant to measure closeness to a singular configuration. The minimum singular value is a possible metric, but it can have a discontinuous slope because of its definition. To circumvent this problem, we chose the product of the singular values as our dynamic singularity measure, normalized against the maximum possible value over all configurations, to give a value

between zero and one. The measure is represented by c , and is computed by

$$c = \frac{\prod_{i=1}^3 \sigma_i(\hat{J}_2(\theta))}{\max_{\theta} \left(\prod_{i=1}^3 \sigma_i(\hat{J}_2(\theta)) \right)} \quad (280)$$

Using the new trajectory, the CSA controller was simulated with various outer loop gains. Because of the greatly increased size of the position step compared to the earlier trajectory, the CMGs quickly approached a singular configuration even when a relatively low gain ($K_1 = 0.10$) was used. By design, the controller then shifts to a nearly free-floating mode of operation. Since the outer loop gain affects how quickly the CMGs saturate, it also determines how quickly the task must be shifted primarily to the arm joints. As the system is forced to move the end-effector over greater distances in a quasi-free-floating mode, the likelihood of encountering dynamic singularities increases. Eventually, this places an upper limit on the outer loop gain that can be used in the controller for this maneuver. Figures 53 and 54 show the dynamic singularity measure and end-effector response for selected outer loop gains. For gains greater than $K_1 = 0.185$, the system encountered a dynamic singularity at about $t = 15$ seconds, causing the arm joint velocity commands to approach infinity.

To solve the problem of simultaneous CMG and dynamic singularities, we recall the two basic options noted for avoiding CMG singularities: slowing the response and emulating a free-floating SMM. The preferred option was to phase in the free-floating mode by increasing the scale factor on the second term of Eq. (275), thus preserving the response time. However, when a dynamic singularity is approached, the second term explodes. This is unacceptable, so the logical alternative is to "turn off" the second term when near a dynamic singularity. In this case, we must resort to the other option of avoiding the CMG singularity, slowing the response time. The system response behavior can be varied dynamically by adding a new scale factor to the first term in Eq. (275) as

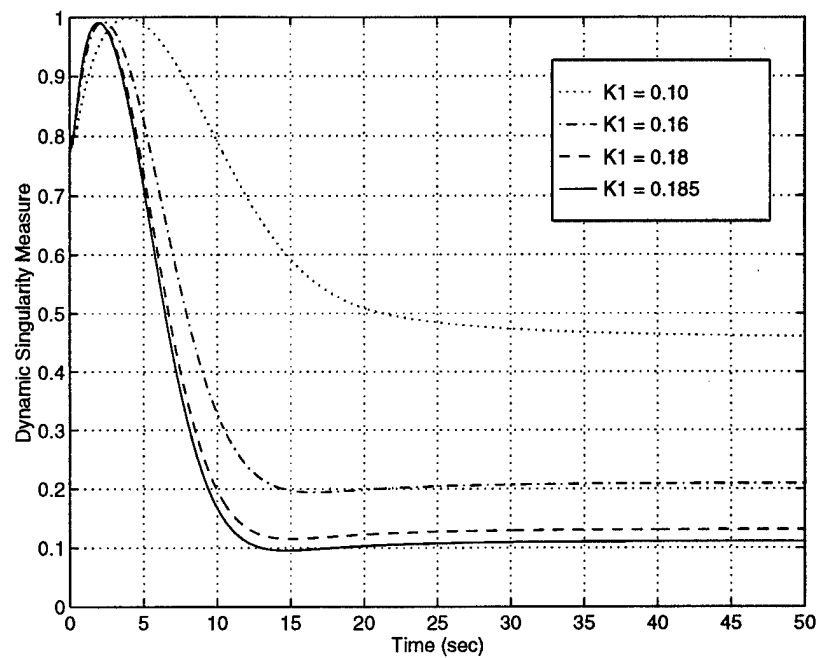


Figure 53. Dynamic Singularity Measure For Varied Outer-Loop Gains Using the CMG Singularity Avoidance Controller

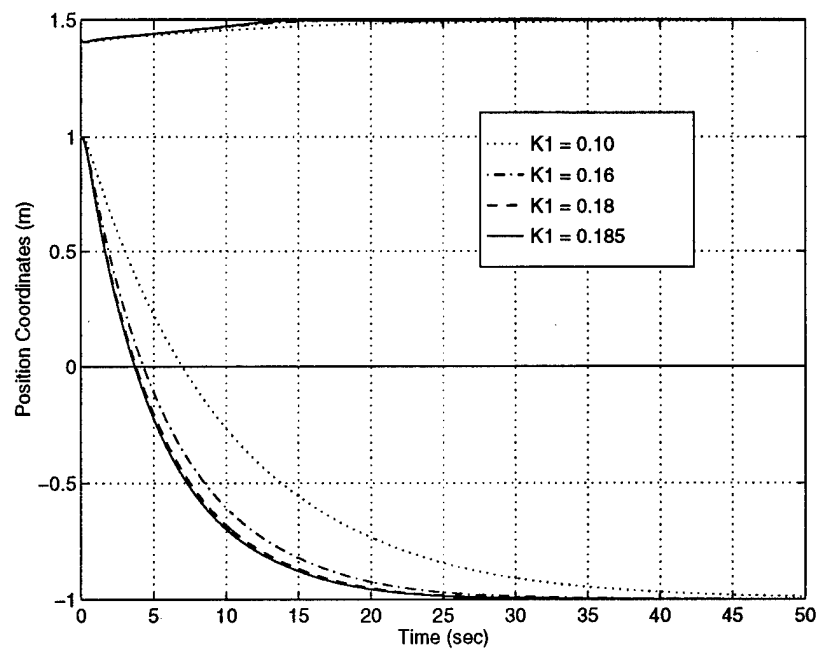


Figure 54. End-Effector Response For Varied Outer-Loop Gains Using the CMG Singularity Avoidance Controller

Dynamic Singularity	CMG Singularity	
	Near	Far
Near	k_1 LOW k_2 LOW	k_1 HIGH k_2 LOW
Far	k_1 HIGH k_2 HIGH	k_1 HIGH k_2 LOW

Table 11. Discrete Function For Scale Factors vs.Singularity States

well, giving

$$\dot{q} = k_1 \dot{q}_1 + k_2 \dot{q}_2 \quad (281)$$

The values for the scale factors can be visualized for four discrete states of the system, based on the nearness of the system to CMG and dynamic singularities, as shown in Table 11. Note the nominal state, far from all singularities, is to use CMG cluster and arm joints equally, which is represented by the first term of Eq. (275).

Rather than incorporate discrete decisions into the continuous controller, the following algorithm was generated to determine \dot{q}_c from \dot{r}_c . First, the basic singularity measures c and d are computed using Eqs. (280) and (278). Then these basic measures are shaped to provide the desired discrete-like form using the formulas

$$c^* = \left(\frac{c}{c + \varepsilon} \right)^{a/(c+\varepsilon)} \quad (282)$$

$$d^* = \left(\frac{d}{d + \varepsilon} \right)^{a/(d+\varepsilon)} \quad (283)$$

where we use the same constants as in Eq. (279). The scale factors are then generated from

$$k_1 = c^* + d^* \quad (284)$$

$$k_2 = (1 - d^*) c^* \quad (285)$$

Finally, the joint velocity commands are used by substituting these scale factors into Eq. (281). This algorithm enforces the general logic of Table 11 while providing a continuous controller. We term this controller the simultaneous singularity avoidance (SSA) controller.

To demonstrate how the SSA controller works, we consider the second example trajectory again. Using the SSA controller, the outer loop gain can be increased beyond the highest value attainable using the CSA controller. The resulting end-effector response is shown in Figure 55, and the corresponding dynamic singularity measure is given in Figure 56. Examining the end-effector response, we see that the larger outer loop gain used in the SSA controller causes the system to approach a dynamic singularity very fast. At about $t = 5$ seconds, the system reaches the point at which the free floating mode begins to shut off. This stops the decline of the dynamic singularity measure, but because the CMG is still nearly singular as well, it also induces a marked change in the end-effector response. This is clearly seen in the end-effector x -coordinate, which changes slope dramatically at this time. The overall response time of the new controller is not much changed from the previous controller, even though it uses a significantly higher outer loop gain. In both cases, the end-effector reaches the final position at about $t = 30$ seconds, suggesting that there is an upper limit to the performance of the system using an RMRC type of control. In order to improve the performance, a CMG cluster with larger control authority would be required. This would delay the point at which the CMG cluster saturates, allowing both faster base angular velocity and faster joint motion.

Although the SSA controller did not produce a significantly better response in this example, it is preferable to the CSA controller because of its flexibility. Given a wide range of possible maneuvers, the CSA controller gains would have to be set low enough to avoid dynamic singularities in the worst case. However, the SSA controller gain could be safely set much higher. This would improve performance of less demanding maneuvers without risking an encounter with dynamic singularities in large maneuvers.

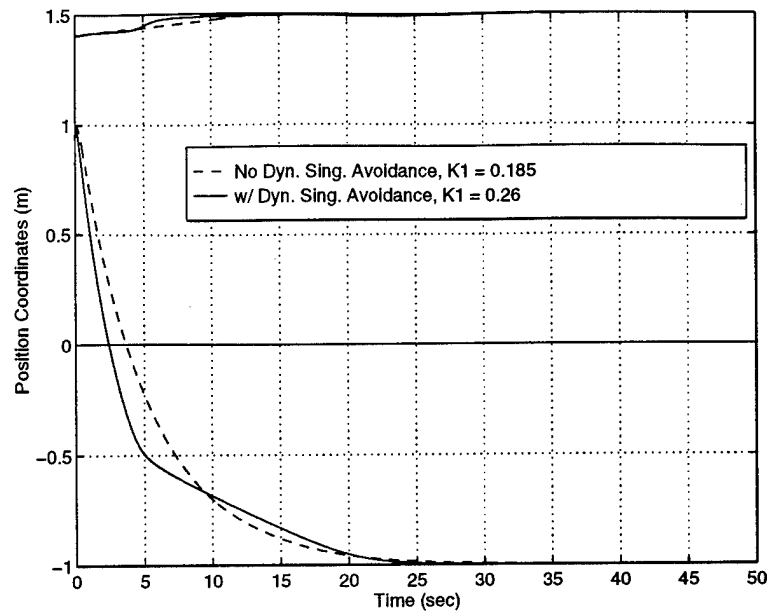


Figure 55. End-Effector Response Using the SSA (w/ Dynamic Singularity Avoidance) Controller and the CSA (No Dynamic Singularity Avoidance) Controller

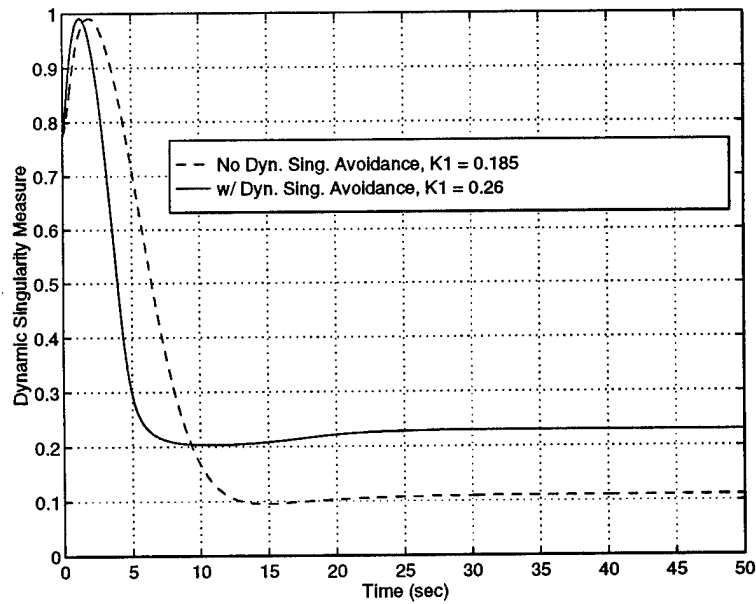


Figure 56. Dynamic Singularity Measure Using the SSA (w/ Dynamic Singularity Avoidance) Controller and the CSA (No Dynamic Singularity Avoidance) Controller

6.4 Summary

In this chapter, we incorporated a cluster of single-gimbal control moment gyros into our dynamic model and into the basic SMM controller. We studied the properties of CMG cluster singularities in nonredundant clusters, finding them to be functionally equivalent to a state of actuator saturation. By using the redundancy inherent in a base-attitude controlled SMM, we developed a controller, termed the CSA controller, that could avoid these cluster singularities. This controller differed from the RBTC controller of Chapter 5, in that it fully controlled the base attitude under most conditions, and phased in a free-floating mode only to alleviate peak torque requirements on the base. By smoothing these peaks, the CMG singularities were avoided. Finally, we noted that in large maneuvers, this method of avoiding gyro singularities could result in simultaneous encounters with dynamic and cluster singularities. Another variation of the SMM controller, termed the SSA controller, solved this problem by adaptively changing the gain on the primary task, slowing the system response when simultaneous singularity conditions occur.

Chapter 7 - Conclusion

7.1 Conclusions

Throughout this work, the primary focus has been to demonstrate the need for base control in an SMM system, and to develop viable base control concepts. Chapters 3-6 have each contributed to this goal, blending the work of earlier researchers with new results and findings to various extents. In this chapter, the main points of base control argument and the significant contributions of this dissertation are highlighted.

Chapter 3 laid the foundation for study of SMM control concepts. The equations of motion and expressions for the system momenta were developed for a system consisting of rigid bodies. This derivation of the equations of motion was the first to incorporate the base angular velocity using the quasi-coordinate formulation of Lagrange's Equation. This method is more flexible than earlier derivations, as it is applicable to spatial representations of both the free-floating and base controlled cases. By combining the equations with a Euler parameter representation of the attitude, additional singularity problems were avoided.

Chapter 4 began with a description of some key concepts from other authors, including the Generalized Jacobian Matrix, dynamic singularities, and Path Dependent Workspace. The GJM was developed as the free-floating extension of the fixed-base manipulator Jacobian using conservation of momentum to eliminate the base motion variables of the free-floating SMM. Joint configurations at which the GJM become singular were termed dynamic singularities. It was shown that these singularities could be associated with large regions of workspace, termed Path Dependent Workspace, where dynamic singularities could be possibly encountered depending on the path of the SMM.

A contribution of Chapter 4 was the exploration of alternative designs which were shown to alleviate problems with dynamic singularities. It was shown that adding redundant degrees of

freedom to the arm increased the PDW, but enabled a local method of singularity avoidance. This method consisted of using null motion to decrease a potential function designed to have maxima at the singular configurations. Prismatic joints and joint limits were both shown to reduce the PDW by eliminating singularity sets from joint space without affecting reachable workspace. A design using a combination of redundancy, prismatic joints, and joint limits was suggested which completely eliminated dynamic singularities from the workspace. Although this design offered advantages over a simple nonredundant, revolute design, it was not ideal because the system could still be trapped at a prismatic joint limit.

Finally, the effect of base control was investigated. It was shown for the general case that total base control eliminates all Jacobian singularities. For base attitude control, examples were given to demonstrate that singularities are essentially equivalent to kinematic singularities in the sense that no region of PDW is created. However, it was also shown that the singular configurations are not necessarily identical to the singular configurations predicted by kinematics alone, but are still dependent on the inertial properties of the system.

In Chapter 5, the comparison of the free-floating and base-attitude control concepts was extended by evaluating their performance in simulated maneuvers. A new SMM controller was introduced which could be modified for use with a variety of control concepts. Modifications principally involved changing the method of choosing joint space velocities from task space velocity commands, from simple matrix inversion of the fixed-base manipulator Jacobian to task priority methods using the pseudoinverse and a null space parameterization of all solutions. Simulations demonstrated that momentum constrained Jacobians like the GJM provide better tracking than the fixed base manipulator Jacobian, even when base motion feedback is used. The controller variation which used the GJM was shown to fail when dynamics singularities were encountered in a simple maneuver inside the PDW. Free-floating control approaches which do not depend on a Jacobian in-

verse have been suggested by other authors, but these methods cannot follow precise paths between points. It is demonstrated that by controlling base attitude, dynamic singularities are eliminated and precise path following is possible throughout the reachable workspace. Although fuel costs can be a serious disadvantage of the base attitude control concept if implemented with thrusters, the Reduced Base-Torque Controller is shown to greatly reduce these costs. The RBTC is designed to follow a nearly free-floating trajectory away from dynamic singularities, and switch to active base control when dynamic singularities are encountered.

Controlling base attitude with control moment gyroscopes is considered in Chapter 6. The development in this chapter is the first to incorporate a CMG cluster into the dynamic model of an SMM. It was shown that for a nonredundant cluster, CMG singularities can be treated as a state of actuator saturation. A new controller is introduced based on this idea, avoiding CMG singularities by shifting towards a free-floating mode to alleviate demands on the cluster when it approaches saturation. Another controller variation builds on this controller, adding a method that avoids simultaneous singularities of the CMG cluster and the SMM system by slowing response time when necessary.

In consideration of the analyses and simulations presented in this work, it is apparent that free-floating control is not always a satisfactory option for Satellite-Mounted Manipulators. Using current techniques, the free-floating SMM must choose between avoiding dynamic singularities and precise tracking. However, by adopting a base attitude control concept, dynamic singularities are eliminated and precise tracking is possible over the entire workspace. The inherent redundancy of a base attitude controlled SMM can be further exploited to minimize costs associated with base actuation.

7.2 Recommendations for Future Research

The development of the SMM controller included several key assumptions which limit its application to some extent. First, a perfect knowledge of the inertial properties of the system was assumed. This is reasonable for the free motion (not in contact with the environment) of the system, although the robustness of the controller to small parameter changes was not investigated. However, when the system comes in contact with its environment, such as when the end-effector grasps an object, the dynamics change significantly and some method of identifying the new inertial parameters and adaptively changing the controller may be essential.

The second significant limitation of the work stems from neglecting flexible motion of the manipulator links. The realities of spacecraft design suggest that links on a space robot will be lightweight and flexible. It is assumed that the flexible effects can be minimized by slowing the motion of the system, but this work did not determine the extent to which this reduction in velocity or acceleration was needed, or how robust the controller would be with respect to these effects. An important first step in this direction would be to add a flexible mode to the links in the system model and explore the controller performance over a variety of response times for different maneuvers.

Neglecting flexibility led to the final major assumption in the controller development. Since flexible motion was not explicitly addressed, it was assumed that all desired trajectories would be planned with low acceleration. Thus, the command joint accelerations were assumed to be zero ($\ddot{\theta}_c \equiv 0$). This assumption was used in proving the stability of the nonlinear control law. If more extensive simulations or experiments should discover instabilities due to this assumption, it could be relaxed by adding an acceleration term to the feedforward portion of the control law. This was not done in this work to avoid the cost of computing the Jacobian derivative in the controller.

These assumptions and limitations suggest several opportunities for future study in the area of SMM control. These include:

- *Adaptive control*: Some method of identifying unknown inertial properties in the system and appropriately adapting the controller would improve the free movement of the end-effector. It may be essential to do this when grasping an object.
- *Optimal control*: The controllers suggested in this work improve system performance in terms of the given metrics, but do not produce true global optimal performance. Applying optimal control methods could result in significant improvements. A possible method of reducing the dimension of the problem would be to search for the optimal scale factor history of the zero-angular momentum term (free-floating mode term) in the RBTC or CSA controllers.
- *New metrics*: The performance metrics chosen here were arbitrarily simple. Other metrics may exist that could be used with the same controller concepts to improve performance.
- *Free-floating mode with non-zero angular momentum*: The controller concepts investigated in this work assumed a free-floating mode with zero angular momentum. The effect of a non-zero constraint over a portion of a maneuver could affect the singularities of the system and be used to improve performance.
- *Redundant CMG cluster*: Only nonredundant CMG clusters were addressed in this work. Incorporating redundancy would introduce interior CMG singularities which would require a different method of avoidance. Techniques to avoid interior singularities using the combined redundancy of the cluster and the arm should be investigated.
- *Physical Experiments*: The final recommendation must acknowledge the importance of practical physical experiments. The conclusions presented here rely heavily on experience

with a simulated system, which inherently includes many assumptions and simplifications. While space testing may be prohibitively expensive at this stage of SMM research, some real experiments using the control concepts of this dissertation would be desirable.

APPENDIX A - Some Notes on Notation

There are several notational devices and conventions in the dissertation that may not be universally understood. The following explanations should clarify them somewhat. In referring to the position or velocity variables, the frame of reference is critical. In this work, a combination of superscripts and subscripts is used to distinguish the reference frames associated with a given variable. The clearest way to explain their use is by example. A position variable, r_a^b , indicates the position of a point fixed in the b -frame (typically its origin) with respect to the a -frame. Similarly, an angular velocity variable, ω_a^b , is the angular velocity of the b -frame with respect to the a -frame. Rotation matrices are handled in much the same way. R_a^b denotes the rotation of the b -frame with respect to the a -frame, so that if a vector \mathbf{v} has components v in the b -frame, its components in the a -frame are given by $R_a^b v$.

Although many of the fundamental quantities of dynamics are best represented by vector expressions, matrix expressions are often better suited for computations. One notational device often used in the translation is the $(\dots)^\times$ operator. This operator is used to designate a skew-symmetric matrix formed from vector components in the following way: Given a vector \mathbf{v} with components $v = (v_1, v_2, v_3)$, v^\times is defined by

$$v^\times = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \quad (286)$$

This operator is used most often when translating a vector cross-product to a matrix representation.

A cross product, $\mathbf{v} \times \mathbf{w}$ becomes $v^\times w$. Some useful properties associated with the operator are:

$$\begin{aligned} v^\times w &= -w^\times v \\ (v^\times)^\top &= -v^\times \end{aligned} \quad (287)$$

Another notation used in some of the developments of Chapter 3 is index notation. This notation is convenient for situations in which higher order tensors arise, making matrix notation insufficient or awkward. A complete description of this notation would be rather lengthy, but the most important features are described here. In index notation, there are two types of indices: dummy indices and free indices. A dummy index is an index that is repeated within a single term of an expression, and implies a summation over that index. For example, given two vectors \mathbf{a} and \mathbf{b} , their dot product can be represented by $a_i b_i$. This is shown below,

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^T \mathbf{b} = a_i b_i = \sum_{i=1}^3 a_i b_i$$

Generally, the summation limits are to be understood from context. In contrast to the vector example above, in this paper the summation is most often over the number of links in the robot. Note that an index may only be repeated one time in a term. A term in which a given index appears three or more times is not defined. A free index is an index that appears only once in a given term and the number of free indices indicates of what order tensor that element is a part. For example, one free index in a term means that the term is an element of a vector (1^{st} order tensor), while two free indices means it is an element of a matrix (2^{nd} order tensor). In any expression, the number of free indices in each term must be equal. Some examples of index notation are:

Example 1 *Matrix-Vector multiplication:*

$$Ab \implies (Ab)_i = A_{ij} b_j$$

Example 2 *Matrix-Matrix multiplication:*

$$AB \implies (AB)_{ij} = A_{ik} B_{kj}$$

The primary reason for index notation is its elegant handling of 3^{rd} order tensors, which appear occasionally in this work. For example, consider a matrix, $M \in \mathbb{R}^{n \times n}$, which is a function of a vector, $x \in \mathbb{R}^n$. The derivative of M with respect to x is a tensor, whose elements require three indices. Manipulation of this tensor is straightforward with index notation, as shown below

Example 3 *3rd Order Tensor-Vector multiplication:*

$$\frac{d}{dt}(M(x(t))) = \overbrace{\frac{\partial M}{\partial x}}^{3^{\text{rd}} \text{ order tensor}} \dot{x}$$

$$\frac{\partial M}{\partial x} \dot{x} \Rightarrow \dot{M}_{ij} = \frac{\partial M_{ij}}{\partial x_k} \dot{x}_k$$

Two useful functions when working with index notation are the Kronecker delta, δ_{ij} , and the alternator function, e_{ijk} . The Kronecker delta is defined by

$$\delta_{ij} \triangleq \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases}$$

The alternator is defined by

$$e_{ijk} \triangleq \begin{cases} 1 & \text{for } ijk = 123, 312, 231 \\ -1 & \text{for } ijk = 213, 321, 132 \\ 0 & \text{repeated subscripts} \end{cases}$$

APPENDIX B - Useful Identities and Properties

B.1 Differentiating the Base Rotation Matrix

The position of a point fixed in the spacecraft base with respect to the \mathcal{F}_I frame can be considered the sum of the position of the base center of mass and a position relative to this point. In vector form,

$$\mathbf{r}_p = \mathbf{r}_0 + \mathbf{r}_0^p \quad (288)$$

Differentiating this *vector* equation with respect to the inertial frame gives

$$\dot{\mathbf{r}}_p = \dot{\mathbf{r}}_0 + \dot{\mathbf{r}}_0^p + \boldsymbol{\omega}_0 \times \mathbf{r}_0^p \quad (289)$$

If we assume that the base does not translate, $\dot{\mathbf{r}}_0 = 0$, and since the point is fixed in the base, $\dot{\mathbf{r}}_0^p = 0$.

Then Eq. (289) becomes

$$\dot{\mathbf{r}}_p = \boldsymbol{\omega}_0 \times \mathbf{r}_0^p \quad (290)$$

Converting to matrix form, Eq. (290) can be written as

$$\dot{r}_p = R_I^0 \boldsymbol{\omega}_0^\times r_0^p \quad (291)$$

where we have written \dot{r}_p in inertial frame (\mathcal{F}_I) components and $\boldsymbol{\omega}_0$ and r_0^p in base frame (\mathcal{F}_0) components. The rotation matrix, R_I^0 , converts \mathcal{F}_0 components to \mathcal{F}_I components. Now suppose that Eq. (288) is converted to matrix form before differentiating. It becomes

$$r_p = r_0 + R_I^0 r_0^p \quad (292)$$

where we have written r_p and r_0 in \mathcal{F}_I components and r_0^p in \mathcal{F}_0 components. Now differentiating the *matrix* expression, Eq. (292), gives

$$\dot{r}_p = \dot{r}_0 + \dot{R}_I^0 r_0^p + R_I^0 \dot{r}_0^p \quad (293)$$

As before, \dot{r}_0 and \dot{r}_0^p are both zero, leaving

$$\dot{r}_p = \dot{R}_I^0 r_0^p \quad (294)$$

Equating the left-hand side of Eqs. (291) and (294), we have

$$\dot{R}_I^0 r_0^p = R_I^0 \omega_0^\times r_0^p \quad (295)$$

Since this is true for any arbitrary point, r_0^p , in the base, we must have

$$\dot{R}_I^0 = R_I^0 \omega_0^\times \quad (296)$$

B.2 Reducing the Equations of Motion

The equations of motion for the SMM are originally written with a full set of generalized coordinates as

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} = Q \quad (297)$$

In the derivation of the SMM controller, it is convenient to reduce the equations to include only the actuated coordinates, eliminating the equations and coordinates q_i for which $Q_i \equiv 0$. First, divide Eq. (297) into two sets of equations, as

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{pmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{pmatrix} + \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \end{pmatrix} = \begin{pmatrix} 0 \\ Q_2 \end{pmatrix} \quad (298)$$

Solve the top equation for \ddot{q}_1 ,

$$\ddot{q}_1 = -M_{11}^{-1} (M_{12} \ddot{q}_2 + C_{11} \dot{q}_1 + C_{12} \dot{q}_2) \quad (299)$$

Substitute this result into the lower equation,

$$\begin{aligned} (M_{22} - M_{21} M_{11}^{-1} M_{12}) \ddot{q}_2 + (C_{21} - M_{21} M_{11}^{-1} C_{11}) \dot{q}_1 \\ + (C_{22} - M_{21} M_{11}^{-1} C_{12}) \dot{q}_2 = Q_2 \end{aligned} \quad (300)$$

Now \dot{q}_1 is eliminated using the momentum constraint,

$$M_{11} \dot{q}_1 + M_{12} \dot{q}_2 = 0 \quad (301)$$

$$\dot{q}_1 = -M_{11}^{-1} M_{12} \dot{q}_2 \quad (302)$$

so that Eq. (300) becomes

$$\tilde{M} \ddot{q}_2 + \tilde{C} \dot{q}_2 = Q_2 \quad (303)$$

where

$$\tilde{M} = M_{22} - M_{21}M_{11}^{-1}M_{12} \quad (304)$$

$$\tilde{C} = C_{22} - M_{21}M_{11}^{-1}C_{12} - (C_{21} - M_{21}M_{11}^{-1}C_{11})M_{11}^{-1}M_{12} \quad (305)$$

B.3 Passivity property applied to SMMs

In the discussion of the stability of the SMM controller developed in Chapter 5, it was assumed that the reduced equations of motion had the following property:

$$\dot{\theta}^\top (\dot{\tilde{M}} - 2\tilde{C}) \dot{\theta} = 0 \quad (306)$$

An analogous expression for terrestrial robot manipulators is referred to as the “passivity” property by Murray, Li, and Sastry [27], and can be shown to be a direct result of the conservation of energy.

In this appendix, we offer a proof that demonstrates that this property applies to SMMs as well.

For an SMM, the total energy is assumed to be equal to the total kinetic energy, which can be written in two equivalent forms,

$$T = \frac{1}{2} \dot{q}^\top M(q) \dot{q} \quad (307)$$

or

$$T = \frac{1}{2} \dot{q}_2^\top \tilde{M}(q) \dot{q}_2 \quad (308)$$

Differentiating $T = T(q, \dot{q})$ with respect to time gives

$$\dot{T} = \frac{\partial T}{\partial q} \dot{q} + \frac{\partial T}{\partial \dot{q}} \ddot{q} \quad (309)$$

Now rearranging Lagrange's Equation provides the relation

$$\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}} \right) - \frac{\partial T}{\partial q} = Q^\top \quad (310)$$

$$\frac{\partial T}{\partial q} = \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}} \right) - Q^\top \quad (311)$$

Substituting Eq. (311) into Eq. (309),

$$\dot{T} = \frac{\partial T}{\partial \dot{q}} \ddot{q} + \left(\frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}} \right) - Q^\top \right) \dot{q} \quad (312)$$

$$= \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}} \dot{q} \right) - Q^\top \dot{q} \quad (313)$$

$$= \frac{d}{dt} (\dot{q}^\top M \dot{q}) - Q^\top \dot{q} \quad (314)$$

$$= 2\dot{T} - Q^\top \dot{q} \quad (315)$$

Then

$$\dot{T} = Q^\top \dot{q} \quad (316)$$

Now from the previous section of this appendix, the equations of motion can be written in a reduced form, in which unactuated coordinates are designated by q_1 and the actuated coordinates are designated by q_2 . Using this notation,

$$\dot{T} = Q_1^\top \dot{q}_1 + Q_2^\top \dot{q}_2 = Q_2^\top \dot{q}_2 \quad (317)$$

since $Q_1 \equiv 0$ by definition.

Now differentiating Eq. (308) gives

$$\dot{T} = \dot{q}_2^\top \tilde{M} \ddot{q}_2 + \frac{1}{2} \dot{q}_2^\top \dot{\tilde{M}} \dot{q}_2 \quad (318)$$

and from the reduced equations of motion, Eq. (303),

$$\tilde{M} \ddot{q}_2 = Q_2 - \tilde{C} \dot{q}_2 \quad (319)$$

Substituting this into Eq. (318) and equating the result to Eq. (317) gives

$$\dot{T} = \dot{q}_2^\top (Q_2 - \tilde{C} \dot{q}_2) + \frac{1}{2} \dot{q}_2^\top \dot{\tilde{M}} \dot{q}_2 = Q_2^\top \dot{q}_2 \quad (320)$$

$$= \dot{q}_2^\top Q_2 + \dot{q}_2^\top \left(\frac{1}{2} \dot{\tilde{M}} - \tilde{C} \right) \dot{q}_2 = Q_2^\top \dot{q}_2 \quad (321)$$

Recalling that \dot{T} is a scalar, each term is equal to its transpose, so subtracting $Q_2^\top \dot{q}_2$ from each side leaves

$$\dot{q}_2^\top \left(\frac{1}{2} \dot{\tilde{M}} - \tilde{C} \right) \dot{q}_2 = 0 \quad (322)$$

or

$$\dot{q}_2^\top \left(\dot{\tilde{M}} - 2\tilde{C} \right) \dot{q}_2 = 0 \quad (323)$$

APPENDIX C - Elements of M and C Matrices

The elements of the M and C matrices used in the equations of motion in this work can be found using the equations in Chapter 3. This process is quite tedious for even the smallest systems and can be done best by a symbolic math software package such as *Mathematica*. For the planar two-link SMM used in much of this work, the elements are shown below.

$$M(1,1) = m_0 + m_1 + m_2$$

$$M(1,2) = 0$$

$$M(1,3) = -a_0 m_1 s_0 - a_0 m_2 s_0 - (a_1 m_1 s_{01})/2 - a_1 m_2 s_{01} - (a_2 m_2 s_{012})/2$$

$$M(1,4) = -(a_1 m_1 s_{01} + 2a_1 m_2 s_{01} + a_2 m_2 s_{012})/2$$

$$M(1,5) = -(a_2 m_2 s_{012})/2$$

$$M(2,2) = m_0 + m_1 + m_2$$

$$M(2,3) = a_0 m_1 c_0 + a_0 m_2 c_0 + (a_1 m_1 c_{01})/2 + a_1 m_2 c_{01} + (a_2 m_2 c_{012})/2$$

$$M(2,4) = (a_1 m_1 c_{01})/2 + a_1 m_2 c_{01} + (a_2 m_2 c_{012})/2$$

$$M(2,5) = (a_2 m_2 c_{012})/2$$

$$M(3,3) = I_0 + I_1 + I_2 + (a_1^2 m_1)/4 + a_0^2 m_1 + a_1^2 m_2 + (a_2^2 m_2)/4 + a_0^2 m_2 + a_1 a_0 m_1 c_1 + 2a_1 a_0 m_2 c_1 + a_1 a_2 m_2 c_2 + a_2 a_0 m_2 c_{12}$$

$$M(3,4) = I_1 + I_2 + (a_1^2 m_1)/4 + a_1^2 m_2 + (a_2^2 m_2)/4 + (a_1 a_0 m_1 c_1)/2 + a_1 a_0 m_2 c_1 + a_1 a_2 m_2 c_2 + (a_2 a_0 m_2 c_{12})/2$$

$$M(3,5) = I_2 + (a_2^2 m_2)/4 + (a_1 a_2 m_2 c_2)/2 + (a_2 a_0 m_2 c_{12})/2$$

$$M(4,4) = I_1 + I_2 + (a_1^2 m_1)/4 + a_1^2 m_2 + (a_2^2 m_2)/4 + a_1 a_2 m_2 c_2$$

$$M(4,5) = I_2 + (a_2^2 m_2)/4 + (a_1 a_2 m_2 c_2)/2$$

$$M(5,5) = I_2 + (a_2^2 m_2)/4$$

$$C(1,1) = 0$$

$$C(1, 2) = 0$$

$$C(1, 3) = -(2a_0m_1c_0\omega + 2a_0m_2c_0\omega + a_1m_1c_0\omega + 2a_1m_2c_0\omega + a_2m_2c_{012}\omega + a_1m_1c_0\dot{\theta}_1 + 2a_1m_2c_0\dot{\theta}_1 + a_2m_2c_{012}\dot{\theta}_1 + a_2m_2c_{012}\dot{\theta}_2)/2$$

$$C(1, 4) = -(a_1m_1c_0\omega + 2a_1m_2c_0\omega + a_2m_2c_{012}\omega + a_1m_1c_0\dot{\theta}_1 + 2a_1m_2c_0\dot{\theta}_1 + a_2m_2c_{012}\dot{\theta}_1 + a_2m_2c_{012}\dot{\theta}_2)/2$$

$$C(1, 5) = -(a_2m_2c_{012}(\omega + \dot{\theta}_1 + \dot{\theta}_2))/2$$

$$C(2, 1) = 0$$

$$C(2, 2) = 0$$

$$C(2, 3) = -(2a_0m_1\omega s_0 + 2a_0m_2\omega s_0 + a_1m_1\omega s_{01} + 2a_1m_2\omega s_{01} + a_1m_1\dot{\theta}_1 s_{01} + 2a_1m_2\dot{\theta}_1 s_{01} + a_2m_2\omega s_{012} + a_2m_2\dot{\theta}_1 s_{012} + a_2m_2\dot{\theta}_2 s_{012})/2$$

$$C(2, 4) = -(a_1m_1\omega s_{01} + 2a_1m_2\omega s_{01} + a_1m_1\dot{\theta}_1 s_{01} + 2a_1m_2\dot{\theta}_1 s_{01} + a_2m_2\omega s_{012} + a_2m_2\dot{\theta}_1 s_{012} + a_2m_2\dot{\theta}_2 s_{012})/2$$

$$C(2, 5) = -(a_2m_2(\omega + \dot{\theta}_1 + \dot{\theta}_2)s_{012})/2$$

$$C(3, 1) = 0$$

$$C(3, 2) = 0$$

$$C(3, 3) = -(a_1a_0m_1\dot{\theta}_1s_1) - 2a_1a_0m_2\dot{\theta}_1s_1 - a_1a_2m_2\dot{\theta}_2s_2 - a_2a_0m_2\dot{\theta}_1s_{12} - a_2a_0m_2\dot{\theta}_2s_{12}$$

$$C(3, 4) = -(a_1a_0m_1\dot{\theta}_1s_1 + 2a_1a_0m_2\dot{\theta}_1s_1 + 2a_1a_2m_2\dot{\theta}_2s_2 + a_2a_0m_2\dot{\theta}_1s_{12} + a_2a_0m_2\dot{\theta}_2s_{12})/2$$

$$C(3, 5) = -(a_2m_2(a_1\dot{\theta}_2s_2 + a_0\dot{\theta}_1s_{12} + a_0\dot{\theta}_2s_{12}))/2$$

$$C(4, 1) = -(a_1m_1c_0\omega + 2a_1m_2c_0\omega + a_2m_2c_{012}\omega + a_1m_1c_0\dot{\theta}_1 + 2a_1m_2c_0\dot{\theta}_1 + a_2m_2c_{012}\dot{\theta}_1 + a_2m_2c_{012}\dot{\theta}_2)/2$$

$$C(4, 2) = -(a_1m_1\omega s_{01} + 2a_1m_2\omega s_{01} + a_1m_1\dot{\theta}_1 s_{01} + 2a_1m_2\dot{\theta}_1 s_{01} + a_2m_2\omega s_{012} + a_2m_2\dot{\theta}_1 s_{012} + a_2m_2\dot{\theta}_2 s_{012})/2$$

$$C(4, 3) = (a_1 m_1 c_{01} \dot{r}_{0x})/2 + a_1 m_2 c_{01} \dot{r}_{0x} + (a_2 m_2 c_{012} \dot{r}_{0x})/2 + 0.5 a_1 a_0 m_1 \omega s_1 + a_1 a_0 m_2 \omega s_1 - (a_1 a_0 m_1 \dot{\theta}_1 s_1)/2 - a_1 a_0 m_2 \dot{\theta}_1 s_1 + (a_1 m_1 \dot{r}_{0y} s_{01})/2 + a_1 m_2 \dot{r}_{0y} s_{01} - a_1 a_2 m_2 \dot{\theta}_2 s_2 + 0.5 a_2 a_0 m_2 \omega s_{12} - (a_2 a_0 m_2 \dot{\theta}_1 s_{12})/2 - (a_2 a_0 m_2 \dot{\theta}_2 s_{12})/2 + (a_2 m_2 \dot{r}_{0y} s_{012})/2$$

$$C(4, 4) = (a_1 m_1 c_{01} \dot{r}_{0x} + 2a_1 m_2 c_{01} \dot{r}_{0x} + a_2 m_2 c_{012} \dot{r}_{0x} + a_1 a_0 m_1 \omega s_1 + 2a_1 a_0 m_2 \omega s_1 + a_1 m_1 \dot{r}_{0y} s_{01} + 2a_1 m_2 \dot{r}_{0y} s_{01} - 2a_1 a_2 m_2 \dot{\theta}_2 s_2 + a_2 a_0 m_2 \omega s_{12} + a_2 m_2 \dot{r}_{0y} s_{012})/2$$

$$C(4, 5) = (a_2 m_2 c_{012} \dot{r}_{0x})/2 + a_1 a_2 m_2 \omega s_2 + 0.5 a_1 a_2 m_2 \dot{\theta}_1 s_2 - 0.25 a_1 a_2 m_2 \dot{\theta}_2 s_2 + (a_2 a_0 m_2 \omega s_{12})/2 + (a_2 m_2 \dot{r}_{0y} s_{012})/2$$

$$C(5, 1) = -(a_2 m_2 c_{012} (\omega + \dot{\theta}_1 + \dot{\theta}_2))/2$$

$$C(5, 2) = -(a_2 m_2 (\omega + \dot{\theta}_1 + \dot{\theta}_2) s_{012})/2$$

$$C(5, 3) = (a_2 m_2 c_{012} \dot{r}_{0x})/2 + 0.5 a_1 a_2 m_2 \omega s_2 - (a_1 a_2 m_2 \dot{\theta}_2 s_2)/2 + 0.5 a_2 a_0 m_2 \omega s_{12} - (a_2 a_0 m_2 \dot{\theta}_1 s_{12})/2 - (a_2 a_0 m_2 \dot{\theta}_2 s_{12})/2 + (a_2 m_2 \dot{r}_{0y} s_{012})/2$$

$$C(5, 4) = (a_2 m_2 (c_{012} \dot{r}_{0x} - a_1 \dot{\theta}_2 s_2 + a_0 \omega s_{12} + \dot{r}_{0y} s_{012}))/2$$

$$C(5, 5) = (a_2 m_2 c_{012} \dot{r}_{0x})/2 + (a_1 a_2 m_2 \omega s_2)/2 + 0.25 a_1 a_2 m_2 \dot{\theta}_1 s_2 + (a_2 a_0 m_2 \omega s_{12})/2 + (a_2 m_2 \dot{r}_{0y} s_{012})/2$$

APPENDIX D - Implementations of Nakamura's Methods

D.1 Singularity-Robust Inverse (SR Inverse)

This appendix provides an overview of the Singularity-Robust inverse (SR inverse) technique, as it was implemented in Chapter 5. For more details on this inversion method, see Ref. [29]. Many robot controllers require an inversion of the manipulator Jacobian to convert workspace velocity to joint space velocity, as in the equation

$$\dot{\theta} = J^{-1}\dot{r} \quad (324)$$

The purpose of the SR inverse is to eliminate excessive joint velocities which can occur near singularities of the Jacobian. The joint speed reduction is attained by trading exactness of the solution to Eq. (324) for a more feasible, but inexact, solution.

Consider how the typical inversion technique, the pseudoinverse, is computed. First, a singular value decomposition (SVD) is performed on the manipulator Jacobian (J), giving

$$J = U\Sigma V^T \quad (325)$$

where U and V are orthonormal matrices and Σ is a diagonal matrix of singular values,

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & \ddots & \\ 0 & 0 & & \sigma_n \end{bmatrix} \quad (326)$$

ordered such that $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$. The columns of U and V form bases for the work and joint spaces respectively. The pseudoinverse is computed by

$$J^\# = V\Sigma^\#U^T \quad (327)$$

where

$$\Sigma^\# = \begin{bmatrix} \frac{1}{\sigma_1} & 0 & 0 & 0 \\ 0 & \frac{1}{\sigma_2} & 0 & 0 \\ 0 & 0 & \ddots & \\ 0 & 0 & & \frac{1}{\sigma_n} \end{bmatrix} \quad (328)$$

If J is singular, one or more of the diagonal elements of Σ will be zero. In $\Sigma^\#$, the associated elements are defined as zero. In this case, movement in the singular direction is impossible, and the pseudoinverse solution will not generate joint velocities associated with velocity components in the singular workspace direction. However, if J is *nearly* singular, then σ_n is very small and $1/\sigma_n$ is very large. This results in large joint velocities even for small workspace velocities in the direction which is nearly singular. As J approaches a singularity, the joint velocities will approach infinity. For a real system, some cutoff value must exist, below which σ_n is treated as zero. This causes a sharp discontinuity in the solution of Eq. (324).

The SR inverse can be defined in a way similar to Eq. (327),

$$J^* = V \Sigma^* U^\top \quad (329)$$

where

$$\Sigma^* = \begin{bmatrix} \frac{\sigma_1}{\sigma_1^2 + k} & 0 & 0 & 0 \\ 0 & \frac{\sigma_2}{\sigma_2^2 + k} & 0 & 0 \\ 0 & 0 & \ddots & \\ 0 & 0 & & \frac{\sigma_n}{\sigma_n^2 + k} \end{bmatrix} \quad (330)$$

The scale factor k is some small, but finite, value. In this way, as J nears a singularity, σ_n will become small, but the diagonal elements of Σ^* will be near those of $\Sigma^\#$ until σ_n gets near k . At this point, σ_n will gracefully go to zero at the singular configuration. The SR inverse will not give an exact solution to Eq. (324), but the error will be small except when near a singularity. The magnitude of the error will depend on the value of the scale factor k . In implementing the SR method in the example in Chapter 5, we used Eqs. (329) and (330) with a scale factor $k = 0.01$.

Using the SR inverse can occasionally result in the robot getting “trapped” at a singularity. This occurs when the desired motion is in the singular direction as the manipulator nears the singular configuration. The SR inverse solution slows the joint velocities down as the manipulator approaches the singularity, and if no other velocity components exist to drive the manipulator through the singular configuration, it can come to a complete stop. Consider a two-link planar arm, in the singular configuration shown in Figure 57, which is commanded to move in the singular direction (x -direction). The Jacobian at this configuration is

$$J = \begin{bmatrix} 0 & 0 \\ 2 & 1 \end{bmatrix} \quad (331)$$

The SVD of J is

$$J = U\Sigma V^T \quad (332)$$

$$U = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (333)$$

$$\Sigma = \begin{bmatrix} 2.24 & 0 \\ 0 & 0 \end{bmatrix} \quad (334)$$

$$V = \begin{bmatrix} 0.894 & 0.447 \\ -0.447 & 0.894 \end{bmatrix} \quad (335)$$

Then the SR inverse is given by

$$J^* = V\Sigma^*U^T \quad (336)$$

$$= \begin{bmatrix} 0.894 & 0.447 \\ -0.447 & 0.894 \end{bmatrix} \begin{bmatrix} 0.447 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad (337)$$

Since the desired velocity is

$$\dot{r} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (338)$$

the inverse equation gives the joint velocity

$$\dot{\theta} = J^*\dot{r} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (339)$$

so if the system reaches the singular configuration, while the controller is demanding movement in the singular direction, the arm comes to a complete stop.

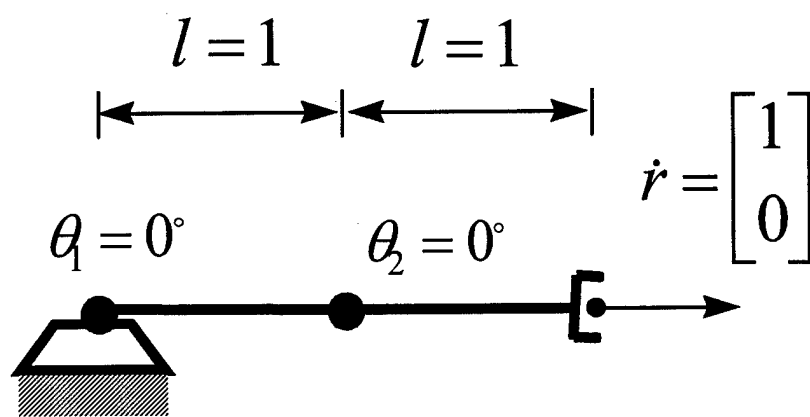


Figure 57. Planar Two-Link In Singular Configuration

D.2 Bidirectional Approach to SMM Path Planning

This appendix outlines the Bidirectional Approach of Nakamura and Mukherjee as it was implemented for the example given in Chapter 5. Further details can be found in Ref. [30]. Their method is slightly modified here, as we did not eliminate the base translation before applying the method. This does not change the results, since the linear momentum constraint is holonomic and the translation coordinate can be included or not without affecting the system dynamics. We chose to retain the translation coordinate purely for convenience, since it made the algorithm more compatible with our existing MATLAB code. In practice, it may be more efficient computationally to eliminate the translation coordinate before applying the bidirectional path planning algorithm. Recall the states of an SMM, defined as base position, base orientation, and arm joint angles

$$x = \begin{bmatrix} r_0 \\ \Omega \\ \theta \end{bmatrix} \quad (340)$$

The state velocities are given by

$$\dot{x} = \begin{bmatrix} \dot{r}_0 \\ \dot{\Omega} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} U & 0 & 0 \\ 0 & A(\Omega) & 0 \\ 0 & 0 & U \end{bmatrix} \begin{bmatrix} v \\ \omega \\ \dot{\theta} \end{bmatrix} \quad (341)$$

Then, using the momentum expressions, Eqs. (101) and (109), a velocity relation can be written between the actuated and unactuated states (recall this method is for a free-floating SMM, so only arm joints are actuated). The momentum expressions, combined into a single matrix equation, are

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} P_v & P_\omega \\ H_v & H_\omega \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} + \begin{bmatrix} P_\theta \\ H_\theta \end{bmatrix} \dot{\theta} \quad (342)$$

Solving for the unactuated base velocities gives

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = - \begin{bmatrix} P_v & P_\omega \\ H_v & H_\omega \end{bmatrix}^{-1} \begin{bmatrix} P_\theta \\ H_\theta \end{bmatrix} \dot{\theta} \quad (343)$$

Combining Eqs. (341) and (343), we can write

$$\dot{x} = G\dot{\theta} \quad (344)$$

where

$$G = \begin{bmatrix} U & 0 & 0 \\ 0 & A(\Omega) & 0 \\ 0 & 0 & U \end{bmatrix} \left[\frac{- \begin{bmatrix} P_v & P_\omega \\ H_v & H_\omega \end{bmatrix}^{-1} \begin{bmatrix} P_\theta \\ H_\theta \end{bmatrix}}{U} \right] \quad (345)$$

Suppose we intend to drive an SMM from some initial state, X_0 , to a desired state X_d . First, construct a virtual system comprised of two identical SMMs, writing a state relation similar to Eq. (344) for the combined system,

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} G_1 & 0 \\ 0 & G_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad (346)$$

Eq. (346) represents a dynamic system that can be controlled to achieve a desired state. Define a Lyapunov function

$$V = \frac{1}{2} \Delta x^\top K \Delta x \quad (347)$$

where K is a positive definite gain matrix (we used an identity matrix), and Δx is the difference between the states of the two halves of the virtual system,

$$\Delta x = x_1 - x_2 \quad (348)$$

Differentiating Eq. (347) gives

$$\dot{V} = \Delta x^\top K (\dot{x}_1 - \dot{x}_2) \quad (349)$$

The system inputs are defined as the actuated joint velocities

$$u_1 \triangleq \dot{\theta}_1 \quad (350)$$

$$u_2 \triangleq \dot{\theta}_2 \quad (351)$$

Then Eq. (349) can be rewritten

$$\dot{V} = \Delta x^\top \hat{K} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (352)$$

where

$$\hat{K} = K \begin{bmatrix} G_1 & -G_2 \end{bmatrix} \quad (353)$$

Now choose the control law

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = -\hat{K}^\# \Delta x \quad (354)$$

so that Eq. (352) becomes

$$\dot{V} = -\Delta x^\top \hat{K} \hat{K}^\# \Delta x \quad (355)$$

Then according to Lyapunov's direct method, the system of Eq. (346) is guaranteed to converge to the state where $\Delta x = 0$, provided that $\dot{V} \leq 0$, with the equality holding only for $\Delta x = 0$. This is true if $\hat{K} \hat{K}^\#$ is positive definite for all states. Unfortunately, Nakamura and Mukherjee show that it is only positive semidefinite, and configurations exist where $\dot{V} = 0$ for $\Delta x \neq 0$. This occurs when \hat{K}^\top has a nontrivial null space, and these configurations are equivalent to dynamic singularities.

Using the control law Eq. (354), the system differential equation, Eq. (346), can be numerically propagated from an initial state

$$x(t_0) = \begin{bmatrix} x_1(t_0) \\ x_2(t_0) \end{bmatrix} = \begin{bmatrix} X_0 \\ X_d \end{bmatrix} \quad (356)$$

to a final state

$$x(t_f) = \begin{bmatrix} x_1(t_f) \\ x_2(t_f) \end{bmatrix} \quad (357)$$

where

$$\Delta x(t_f) = x_1(t_f) - x_2(t_f) = 0 \quad (358)$$

Assuming the system does converge, then a joint trajectory can be constructed for the real system to drive it from X_0 to X_d by combining the two halves of the virtual system as follows:

$$x(t) = \begin{cases} x_1(t) & 0 \leq t \leq t_f \\ x_2(2t_f - t) & t_f \leq t \leq 2t_f \end{cases} \quad (359)$$

To implement this trajectory with our basic SMM controller, the state trajectory of Eq. (359) was converted to an end-effector trajectory using the forward kinematic equation, Eq. (5), giving

$$r(t) = f(x(t)) \quad (360)$$

This end effector trajectory then became the command end-effector position, $r_d(t)$, input to the free-floating controller using the GJM (see Figure 17). Since the end effector trajectory is derived

from a convergent solution of the bidirectional algorithm, dynamic singularities are avoided, and the basic controller performed well.

In our experience with this algorithm, singularities are still frequently encountered if the end state is chosen arbitrarily. However, by choosing end states which correspond to the end states arising from performing the maneuver with the GJM variation of our SMM controller, this method has worked well. In cases like Maneuver Three where the GJM based controller fails due to a singularity, this technique for finding the end state cannot be used. In these instances, adequate results were obtained by using the base attitude at the point where the singularity was encountered and adjusting the joint angles to match the end-effector position to the target point. Another practical concern we have in using this method is the added computational cost not only of the inverse kinematics problem, but also of the path planning algorithm, both of which must be performed before the end-effector can begin moving.

Bibliography

- [1] "Ranger Program Overview." <http://www.ssl.uml.edu>, May 1996.
- [2] Alexander, H.L. and R.H. Cannon. "An Extended Operational-Space Control Algorithm for Satellite Manipulators," *J. of the Astronautical Sciences*, 38 4, 473-486 1990.
- [3] Bedrossian, N.S., Paradiso J. and E.V. Bergmann. "Steering Law Design for Redundant Single-Gimbal Control Moment Gyroscopes," *J. of Guidance, Control, and Dynamics*, 13 6, 1083-1089 1990.
- [4] Chobotov, Vladimir A. *Spacecraft Attitude Dynamics and Control*. Malabar, Florida: Krieger Publishing Co., 1991.
- [5] Conway, B.A. and J.W. Widhalm. "Equations of Attitude Motion for an N-Body Satellite with Moving Joints," *J. of Guidance, Control, and Dynamics*, 8 4, 537-539 1985.
- [6] Denavit, J. and R.S. Hartenberg. "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices," *ASME Trans. J. Appl. Mech.*, 77 215-221 1955.
- [7] Dubowsky, S. and E. Papadopoulos. "The Kinematics, Dynamics, and Control of Free-Flying and Free-Floating Space Robotic Systems," *Trans. on Robotics and Automation*, 9 5, 531-542 1993.
- [8] Faile, G.C., Counter D.N. and E.J. Bourgeois. "Dynamic Passivation of a Spinning and Tumbling Satellite Using Free-Flying Teleoperators." *Proc. Of the 1st Natl. Conf. On Remotely Manned Systems*, edited by E. Heer. 66-73. 1973.
- [9] Ford, K.A. *Reorientations of Flexible Spacecraft Using Momentum Exchange Devices*. PhD dissertation, Air Force Institute of Technology, 1997.
- [10] Ford, K.A. and C.D. Hall. "Flexible Spacecraft Reorientations Using Gimballed Momentum Wheels." Sun Valley, Idaho: AAS/AIAA Astrodynamics Specialist Conference, 1997.
- [11] Hoelscher, B.R. and S.R. Vadali. "Optimal Open-Loop and Feedback Control Using Single Gimbal Control Moment Gyroscopes," *The J. of the Astronautical Sciences*, 42 2, 189-206 1994.
- [12] Hooker, W.W. "A Set of R Dynamical Attitude Equations for an Arbitrary N-Body Satellite Having R Rotational Degrees of Freedom," *AIAA Journal*, 8 7, 1205-1207 1970.
- [13] Hooker, W.W. "Equations of Motion for Interconnected Rigid and Elastic Bodies: A Derivation Independent of Angular Momentum," *Celestial Mechanics*, 11 337-359 1975.
- [14] Hooker, W.W. and G. Marguiles. "The Dynamical Attitude Equations for an N-Body Satellite," *J. of the Astronautical Sciences*, 12 4, 123-128 1965.
- [15] Hughes, P.C. *Spacecraft Attitude Dynamics*. New York, NY: John Wiley & Sons, 1986.
- [16] Krishnan, S. and S.R. Vadali. "Near-Optimal Three Dimensional Rotational Maneuvers of Spacecraft Using Manipulator Arms," *J. of Guidance, Control, and Dynamics*, 18 4, 932-934 1995.
- [17] Liegeois, A. "Automatic Supervisory Control of the Configuration and Behavior of Multibody Mechanisms," *Trans. on Sys., Man., and Cyber.*, 12 868-871 1977.

- [18] Likins, P.W. "Point-Connected Rigid Bodies in a Topological Tree," *Celestial Mechanics*, 11 301-317 1975.
- [19] Lindberg, R.E., Longman R.W. and M.F. Zedd. "Satellite Mounted Robot Manipulators – New Kinematics and Reaction Moment Compensation," *Intl. J. of Robotics Research*, 6 3, 1987.
- [20] Lindberg, R.E., Longman R.W. and M.F. Zedd. "Kinematic and Dynamic Properties of an Elbow Manipulator Mounted on a Satellite," *J. of the Astronautical Sciences*, 38 4, 397-421 1990.
- [21] Longman, R.W. "The Kinetics and Workspace of a Satellite-Mounted Robot," *J. of the Astronautical Sciences*, 38 4, 423-440 1990.
- [22] Luh, J.Y.S., Walker M.W. and R.P.C. Paul. "Resolved-Acceleration Control of Mechanical Manipulators," *IEEE Trans. on Automatic Control*, 25 468-474 1980.
- [23] Luo, Z.H. and Y. Sakawa. "Control of a Space Manipulator for Capturing a Tumbling Object." *Proc. Of the 29th Conf. On Decision and Control*. 103-108. 1990.
- [24] Margulies, G. and J.N. Aubrun. "Geometric Theory of Single-Gimbal Control Moment Gyro Systems," *J. of the Astronautical Sciences*, 26 159-191 1978.
- [25] Meirovitch, L. *Methods of Analytical Dynamics*. McGraw-Hill, 1970.
- [26] Mukherjee, R. and Y. Nakamura. "Formulation and Efficient Computation of Inverse Dynamics of Space Robots," *IEEE Trans. on Robotics and Automation*, 8 3, 400-406 1992.
- [27] Murray, R.M., Li Z. and S.S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.
- [28] Nagamatsu, H., Kubota T. and I. Nakatani. "Capture Strategy for Retrieval of a Tumbling Satellite by a Space Robotic Manipulator." *Proc. Of the 1996 IEEE Intl. Conf. On Robotics and Automation*. 70-75. 1996.
- [29] Nakamura, Y. *Advanced Robotics: Redundancy and Optimization*. Addison-Wesley, 1991.
- [30] Nakamura, Y. and R. Mukherjee. "Nonholonomic Path Planning of Space Robots Via a Bidirectional Approach," *IEEE Trans. on Robotics and Automation*, 7 4, 500-514 1991.
- [31] Nakamura, Y. and R. Mukherjee. "Exploiting Nonholonomic Redundancy of Free-Flying Space Robots," *IEEE Trans. on Robotics and Automation*, 9 4, 499-506 1993.
- [32] Oda, M. "Coordinated Control of Spacecraft Attitude and its Manipulator." *Proc. Of the 1996 IEEE Intl. Conf. On Robotics and Automation*. 732-738. 1996.
- [33] Onega, G.T. and J.H. Clingman. "Free-Flying Teleoperator Requirements and Conceptual Design." *Proc. Of the 1st Natl. Conf. On Remotely Manned Systems*, edited by E. Heer. 19-31. 1973.
- [34] Papadopoulos, E. and S. Dubowsky. "On the Nature of Control Algorithms for Free-Floating Space Manipulators," *IEEE Trans. on Robotics and Automation*, 7 6, 750-758 1991.
- [35] Papadopoulos, E. and S. Dubowsky. "Dynamic Singularities in Free-Floating Space Manipulators," *J. of Dynamic Systems, Measurement, and Control*, 115 44-52 1993.
- [36] Paradiso, J.A. "Global Steering of Single Gimballed Control Moment Gyroscopes Using a

- Directed Search," *J. Guidance, Control, and Dynamics*, 15 5, 1236-1244 1992.
- [37] Quinn, R.D. "Equations of Motion for Structures in Terms of Quasi-Coordinates," *J. of Applied Mechanics*, 745-749 1990.
 - [38] Reyhanoglu, M. and N.H. McClamroch. "Controllability and Stabilizability of Planar Multibody Systems with Angular Momentum Preserving Control Torques." *Proc. Of the American Control Conf.* 1102-1107. 1991.
 - [39] Reyhanoglu, M. and N.H. McClamroch. "Planar Reorientation Maneuvers of Space Multibody Systems Using Internal Controls," *J. of Guidance, Control, and Dynamics*, 15 6, 1475-1480 1992.
 - [40] Saha, S.K. "A Unified Approach to Space Robot Kinematics," *Trans. on Robotics and Automation*, 12 3, 401-405 1996.
 - [41] Spofford, J.R. and D.L. Akin. "Redundancy Control of a Free-Flying Telerobot," *J. of Guidance, Control, and Dynamics*, 13 3, 515-523 1990.
 - [42] Spong, M.W. and M. Vidyasagar. *Robot Dynamics and Control*. New York, NY: John Wiley & Sons, 1989.
 - [43] Umetani, Y. and K. Yoshida. "Continuous Path Control of Space Manipulators Mounted on an OMV," *Acta Astronautica*, 15 12, 981-986 1987.
 - [44] Umetani, Y. and K. Yoshida. "Resolved Motion Rate Control of Space Manipulators with Generalized Jacobian Matrix," *IEEE Trans. on Robotics and Automation*, 5 3, 303-314 1989.
 - [45] Vafa, Z. and S. Dubowsky. "On the Dynamics of Manipulators in Space Using the Virtual Manipulator Approach." *Proc. Of the IEEE Conf. On Robotics and Automation*. 579-585. 1987.
 - [46] Vafa, Z. and S. Dubowsky. "On the Dynamics of Space Manipulators Using the Virtual Manipulator, with Applications to Path Planning," *J. of the Astronautical Sciences*, 38 4, 441-472 1990.
 - [47] Verhulst, F. *Nonlinear Differential Equations and Dynamical Systems*. Springer-Verlag, 1990.
 - [48] Walker, M.W. and L.B. Wee. "Adaptive Control of Space-Based Robot Manipulators," *IEEE Trans. on Robotics and Automation*, 7 828-835 1991.
 - [49] Wee, L.B. and M.W. Walker. "On the Dynamics of Contact Between Space Robots and Configuration for Impact Minimization," *IEEE Trans. On Robotics and Automation*, 9 5, 581-591 1993.
 - [50] Whitney, D.E. "Resolved Motion Rate Control of Manipulators and Human Prostheses," *IEEE Trans. on Man-Machine Systems*, 10 47-53 1969.
 - [51] Yoshida, K. and Y. Umetani. "Control of a Space Free-Flying Robot." *Proc. Of the 29th Conf. On Decision and Control*. 97-102. 1990.
 - [52] Yoshikawa, T. "Analysis and Control of Robot Manipulators with Redundancy." *Robotics Research* edited by M. Brady and R. Paul, 735-747, Cambridge, MA: MIT Press, 1984.

Vita

Nathan Titus was born in Pontiac, Michigan on February 3, 1965. In 1968, his family moved to Guam, where he spent the remainder of his youth, graduating from the High School Division of the Guam Community College in 1982. With the help of an ROTC scholarship, he attended the University of Washington in Seattle, Washington. In 1986, he graduated with a bachelor's degree in Aeronautics & Astronautics and received his commission in the United States Air Force. He began active duty service in Colorado Springs, serving a short time as a staff officer at HQ Air Force Space Command, and then entered training as a satellite planner/analyst officer in 1987. Upon completing the satellite operations training, he helped activate the 3rd Satellite Control Squadron at Falcon AFB. In 1991, he was assigned to the Air Force Institute of Technology, graduating with a master's degree in Astronautical Engineering in 1992. He was subsequently assigned to the USAF Phillips Laboratory at Kirtland AFB, New Mexico, where he worked in the Astrodynamics Division and the Flight Test Branch. In 1995, he returned to AFIT to pursue a doctorate in Astronautical Engineering. He received the Doctor of Philosophy degree in 1998 and headed back to Kirtland AFB.

Permanent Address: 15310 Washington Ave. SW, Tacoma, WA 98498.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE May 1998		3. REPORT TYPE AND DATES COVERED Dissertation
4. TITLE AND SUBTITLE Singularity Avoidance Strategies For Satellite Mounted Manipulators Using Attitude Control			5. FUNDING NUMBERS	
6. AUTHOR(S) Nathan A. Titus, Major, USAF				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology, WPAFB, OH 45433-7655			8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/DS/ENY/98-03	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFRL/VSDVD (Dr. Richard Madison) 3550 Aberdeen Ave. SE Kirtland AFB, NM 87117 Tel. (505) 846-1339			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES Dissertation Advisor: Dr. Curtis H. Spenny				
12a. DISTRIBUTION AVAILABILITY STATEMENT Approved For Public Release; Distribution Unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Control concepts for satellite-mounted manipulators (SMM) are examined. The primary focus is on base-actuated concepts, which eliminate singularity problems associated with free-floating SMMs. A new form of the equations of motion for an n -link SMM is developed using a quasi-coordinate form of Lagrange's Equation. Alternative free-floating SMM designs are presented which eliminate dynamic singularities, but still experience difficulties due to the unactuated base. A new generic SMM controller is developed as a framework for various control concepts with and without base actuation. Momentum-constrained Jacobians are shown to produce better SMM tracking than fixed-base Jacobians, even when base motion feedback is incorporated into the controller. A variation of the generic controller, termed the Reduced Base-Torque Controller (RBTC), is introduced and shown to reduce attitude control costs significantly while retaining the advantages of base control. The RBTC uses a task priority technique, assigning the first priority to end-effector control and secondary priority to maintaining a zero angular momentum state. Finally, the SMM dynamic model and generic controller are modified to include a cluster of control moment gyroscopes (CMG), and the effects of using the cluster for base attitude control in the SMM system are considered. A controller variation is developed which avoids singularities of both the CMG cluster and the manipulator system. A variety of planar and spatial simulations are used to validate the performance of the controllers. The results indicate that the base attitude control concept is the most viable SMM control concept in terms of tracking performance and singularity avoidance.				
14. SUBJECT TERMS Spacecraft; Manipulators; Robots; Robotics; Artificial Satellites; Attitude Control Systems; Satellite Attitude; Satellite Mounted Manipulators; Space Robots; Space Manipulators; Control Moment Gyroscopes			15. NUMBER OF PAGES 188	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	